# NAVAL POSTGRADUATE SCHOOL
# Monterey, California

# THESIS

**EFFECTIVE SPATIALLY SENSITIVE INTERACTION IN VIRTUAL ENVIRONMENTS**

by

Richard S. Durost

September 2000

Thesis Advisor: Rudolph P. Darken
Associate Advisor: Michael Capps

| REPORT DOCUMENTATION PAGE | | | *Form Approved OMB No. 0704-0188* |
|---|---|---|---|
| Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503. | | | |
| **1. AGENCY USE ONLY *(Leave blank)*** | **2. REPORT DATE** September 2000 | **3. REPORT TYPE AND DATES COVERED** Master's Thesis | |
| **4. TITLE AND SUBTITLE:** Title (Mix case letters) Effective Spatially Sensitive Interaction in Virtual Environments | | | **5. FUNDING NUMBERS** |
| **6. AUTHOR(S)** Richard S. Durost, Captain USA | | | |
| **7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)** Naval Postgraduate School Monterey, CA 93943-5000 | | | **8. PERFORMING ORGANIZATION REPORT NUMBER** |
| **9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)** N/A | | | **10. SPONSORING / MONITORING AGENCY REPORT NUMBER** |
| **11. SUPPLEMENTARY NOTES** The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. | | | |
| **12a. DISTRIBUTION / AVAILABILITY STATEMENT** Approved for public release; distribution is unlimited | | | **12b. DISTRIBUTION CODE** |


**13. ABSTRACT *(maximum 200 words)***

Effective interaction techniques are critical for productive use of virtual environments for business, manufacturing, and training. This thesis addresses the need to match the dimensionality of tasks performed in a virtual environment to the dimensionality of the techniques used to perform the tasks.

In order to demonstrate the performance benefits of matching the dimensionality of task and technique, an experiment was conducted in which twenty-seven subjects were asked to perform a series of two and three-dimensional tasks. Subjects were required to perform all tasks using only three-dimensional techniques, then only two-dimensional techniques, and finally a combination of both techniques.

The results clearly showed that matching the dimensionality of the task to the dimensionality of the interaction technique achieved the best performance in a virtual environment. Of 27 subjects, 90% preferred to use a technique whose dimensionality matched the requirements of the task. More importantly, 100% demonstrated improved performance when the dimensionality of task and technique matched.


| **14. SUBJECT TERMS** Virtual Environments, Interaction, Interaction Techniques | | | **15. NUMBER OF PAGES** 154 |
|---|---|---|---|
| | | | **16. PRICE CODE** |
| **17. SECURITY CLASSIFICATION OF REPORT** Unclassified | **18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified | **19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified | **20. LIMITATION OF ABSTRACT** UL |

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

**Approved for public release; distribution is unlimited**

# EFFECTIVE SPATIALLY SENSITIVE INTERACTION IN VIRTUAL ENVIRONMENTS

Richard S. Durost
Captain, United States Army
B.S., United States Military Academy, 1990

Submitted in partial fulfillment of the
requirements for the degree of

**MASTER OF SCIENCE IN COMPUTER SCIENCE**

from the

**NAVAL POSTGRADUATE SCHOOL**
**September 2000**

Author: ______________________________
Richard S. Durost

Approved by: ______________________________
Rudolph P. Darken, Thesis Advisor

______________________________
Michael Capps, Associate Advisor

______________________________
Dan C. Boger, Chairman
Computer Science Academic Group

THIS PAGE INTENTIONALLY LEFT BLANK

# ABSTRACT

Effective interaction techniques are critical for productive use of virtual environments for business, manufacturing, and training. This thesis addresses the need to match the dimensionality of tasks performed in a virtual environment to the dimensionality of the techniques used to perform the tasks.

In order to demonstrate the performance benefits of matching the dimensionality of task and technique, an experiment was conducted in which twenty-seven subjects were asked to perform a series of two and three-dimensional tasks. Subjects were required to perform all tasks using only three-dimensional techniques, then only two-dimensional techniques, and finally a combination of both techniques.

The results clearly showed that matching the dimensionality of the task to the dimensionality of the interaction technique achieved the best performance in a virtual environment. Of 27 subjects, 90% preferred to use a technique whose dimensionality matched the requirements of the task. More importantly, 100% demonstrated improved performance when the dimensionality of task and technique matched.

THIS PAGE INTENTIONALLY LEFT BLANK

# TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

# LIST OF FIGURES

# LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

# LIST OF ACRONYMS

| | |
|---|---|
| 2D | Two Dimensional |
| 3D | Three Dimensional |
| CAVE | CAVE Automated Virtual Environment |
| CHIMP | Chapel Hill Immersive Modeling Program |
| DoD | Department of Defense |
| DOF | Degree of Freedom |
| GUI | Graphical User Interface |
| GVWR | Gross Vehicle Weight Rating |
| HMD | Head-Mounted Display |
| HOMER | Hand-Centered Object Manipulation Extending Ray-casting |
| MAAVE | Multi-Angled Automatic Virtual Environment |
| PDA | Personal Digital Assistant |
| VE | Virtual Environment |
| VEE | Virtual Environment Enclosure |
| VR | Virtual Reality |
| VRML | Virtual Reality Modeling Language |
| WIMP | Window, Icon, Menu, Pointer |

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF TRADEMARKS

CAVE is a trademark of Fakespace Systems, Inc.

Cyrix is a trademark of Cyrix Corp.

Heil is a trademark of Heil Trailer International.

International is a trademark of International Truck and Engine Corp.

Java is a trademark of Sun Microsystems, Inc.

Pentium is a trademark of Intel Corporation.

Peterbilt is a trademark of Peterbilt, Inc.

Polhemus, Fastrak, Long Ranger are trademarks of Polhemus.

Stoughton is a trademark of Stoughton Trailers, Inc.

Trailmobile is a trademark of Trailmobile Corp.

Trailstar is a trademark of Trailstar Manufacturing, Inc.

Vega, VegaNT, LynX, Creator, Multigen are trademarks of Multigen-Paradigm, Inc.

Windows, Windows 95, Windows NT 4.0 are trademarks of Microsoft Corporation.

THIS PAGE INTENTIONALLY LEFT BLANK

## ACKNOWLEDGMENTS

The author wants to thank God for providing the ability and the strength to complete this thesis. He would also like to thank his wife Andrea for her love and support throughout this process, and daughter Katie for understanding when Daddy was tired and had to work late.

The author wants to thank Professors Rudy Darken and Mike Capps for their guidance, patience, and encouragement during the work in developing the experiment and writing this thesis.

THIS PAGE INTENTIONALLY LEFT BLANK

# I. INTRODUCTION

## A. THESIS STATEMENT

Matching the dimensionality of task requirements to interaction techniques will improve performance on tasks as opposed to when a mismatch occurs.

## B. MOTIVATION

The world we live in is, by its nature, inherently 3D. Yet daily we are required to perform tasks that are inherently 2D. Sometimes, the tasks we perform are neither inherently 3D nor 2D, but can be performed using 2D, 3D, or hybrid interaction techniques. It is this dimensional insensitivity of tasks that often presents a dilemma in virtual environment (VE) applications. A VE is, by definition, inherently 3D. Yet we may be required to perform tasks that are inherently 2D within that environment. There are many tools that have been developed to make interaction with the 3D VE relatively simple, just as there are interaction devices that make using a desktop environment intuitive for the user. However, when a task's dimensional requirements are such that they conflict with the inherent dimensionality of the environment in which they must be performed, or when a task is dimensionally insensitive and can be performed using a variety of techniques or devices, an implementation decision is required to enable task performance despite any dimensionality conflict or ambiguity. For this reason, it is clear that *no single interaction technique is optimal for both 2D and 3D tasks.*

This axiom is as true in the realm of 3D virtual environments as it is in the environment we see around us. Virtual environments are an attempt to create a near real, 3D environment with which a user interacts via some form of interface. Media in general and movies in particular have elicited visions of interactive worlds that were never before even thought possible, worlds where people could explore environments that either could never be explored in the real world or that possibly don't even exist. Human imagination even has gone as far as envisioning virtual environments where people could anonymously engage in activities ranging from intellectual encounters to close combat to virtual sex. In actuality, current virtual reality applications are much more restricted in their utility. They are used for tasks such as training, manufacturing, telepresence/telerobotics, and entertainment. However, in the rush to achieve a virtual realm that is limited only by the human imagination, some essential pieces of reality have been left behind. One of these pieces is the fact that many of the tasks we perform in the environment that surrounds us are inherently 2D and therefore demand some form of interface that enables 2D interaction. In current immersive virtual environments, users can perform 3D tasks with relative ease, whereas their ability to accomplish 2D tasks is cumbersome at best, and often non-existent.

Industrial applications of virtual environments provide an apt illustration of this problem. The U.S. Army currently uses a CAVE located in Warren, Michigan at the U.S. Army TACOM National Automotive Center to aid in the development of future Army vehicles such as the Future Fighting Vehicle, the HMMWV with Trailer System, and the Mobile Medical Unit Concept Vehicle. Engineers are able to enter the CAVE and

examine and evaluate development concepts from various perspectives such as ergonomics, functionality, and performance. Unfortunately, some elements of the evaluation process cannot be accomplished within the 3D environment of the CAVE, because there is no means for conducting essential, inherently 2D tasks such as reading specifications on component parts or making annotations about recommended changes or enhancements. For example, in a real world environment, should an engineer evaluating a new component of a system find a new utility for that component or want to suggest a better way of implementing it, the engineer would be able to do something as simple as leaving a sticky note on the part in question, outlining recommendations, criticisms, or suggestions relating to that part. This would enable engineers reviewing the system at a later time to benefit from the input provided. No such capability exists in current VE applications. Instead, engineers must carry paper and pencil into the virtual environment and take notes that later need to be transferred to another form of medium for distribution. The same problem applies when the engineer requires some form of textual output such as a specification document. Instead of being able to access the document from within the CAVE, the engineer must either exit the virtual environment to access the documentation or bring a copy of it into the CAVE.

As the number of applications for virtual environments continues to expand, the need to explore and resolve the problem of 2D interaction within those environments becomes more critical. Research and development of Internet 2 is progressing and shortly universities, corporations, and even individual consumers will begin to have access to capabilities that currently exist only in the research realm. The office of the future is an

example of such an application that will enable companies that are geographically dispersed to meet inside networked virtual environments such as a virtual conference room. Within this environment, executives will be able to conduct face-to-face meetings and perform collaborative work. This interaction will be difficult and potentially ineffective if there is no means for performing the 2D tasks that normally occur within an environment like a conference room.

Virtual environment applications exist in industry that allow immersive collaborative design sessions between groups of geographically separated engineers. Engineers are able to view and manipulate 3D objects within the virtual environment in order to optimize a product design and greatly decrease the time and resource cost associated with its development. They are not, however, able to perform any 2D tasks such as reading or writing within that environment unless they bring items such as paper, pencil, and manuals into the environment with them.

Clearly, with all the advances that have been made in technology and virtual environment research in recent years, one should reasonably expect that some form of 2D computer interaction would be possible in a 3D environment.

## C. RESEARCH QUESTIONS

This research sets out to answer several key questions. Foley (1984, p.21) established six basic types of interaction that occur in all computer applications, regardless of dimensionality. Given that any application can be decomposed into combinations of these six types of interaction, how does one decide whether an

application is best suited for a 2D environment or a 3D environment? If an application requires only purely 2D tasks, it seems obvious that a 2D environment such as a desktop computer would be the best platform for that application. Similarly, if an application consists of purely 3D tasks such as might occur in an architectural walk-through, a 3D environment presented using a CAVE or a Head-Mounted Display (HMD) would probably be a preferable presentation medium. However, when an application requires a combination of 2D and 3D tasks, which is usually the case, there must be some logical process for determining how to best present the application. This research intends to develop an approach that can be used to determine the best presentation medium and associated techniques for implementing an application given its unique dimensional requirements.

The answer to this question leads to yet another question. When an application requires the performance of both 2D and 3D tasks, can the functionality of some tasks be sacrificed to accommodate the dimensional requirements of another? If, for instance, an application consists of predominately 3D tasks, can the 2D tasks be accomplished within a 3D environment using techniques that are not necessarily well suited for 2D tasks? This research seeks to show that although there cannot be one correct answer to this question for all applications, one will be able to more easily arrive at a solution by prioritizing the elementary tasks of an application in terms of their dimensionality.

If the above accommodation cannot be made, how can 2D tasks be performed in an immersive 3D environment? Is there a way to exploit existing technology to enable 2D interaction inside a virtual environment? Although it may seem intuitive that such a

solution exists, there has been little research conducted that addresses this problem. This research will attempt to provide a solution that will enable a user of a virtual environment such as a CAVE to interact with the 3D environment while also providing a means for usable 2D interaction. The 2D interaction devices being used in this research require the user's natural vision and are not intended to be represented virtually. Additionally, current HMD technology does not provide a sufficient combination of visual resolution and field of view such that large amounts of text can be displayed without obscuring the user's view of the surrounding virtual environment. Therefore, no attempt will be made to provide a solution for this problem in environments that are presented using devices such as an HMD that occludes the user's natural vision.

Finally, this research will also propose that a solution to the requirement for 2D and 3D interaction devices and techniques with an immersive 3D environment may be found in the development of hybrid interface, capable of both 2D and 3D interaction.

## D. METHODOLOGY

The following steps were taken in order to answer the questions outlined above:

1. Background Study. Existing 2D, 3D, and hybrid interaction techniques used in VEs were examined in order to further expose the current dilemma that exists when both 2D and 3D interaction is required in a VE.

2. Framework Development. A framework was developed for analyzing the dimensionality of user tasks, the associated interaction technique requirements, and the resulting impact of those requirements on interaction techniques in VE application design.

3. Usability Testing. An experiment was conducted that focused on 2D, 3D, and hybrid interaction techniques in a CAVE virtual warehouse using a Fujitsu Stylistic 1200 tablet, a 3D mouse, and a Polhemus tracking device.

## E. ORGANIZATION OF THESIS

Chapter II contains pertinent background information.

Chapter III provides a description of the framework used to analyze the requirements for 2D/3D interaction techniques in a VE.

Chapter IV describes the methodology used in testing the theory of improved performance when no dimensionality conflicts occur.

Chapter V examines the results and provides an analysis of the performance data collected during the experiment.

Chapter VI contains the conclusions reached and recommendations resulting from the experimental results.

Chapter VII describes potential future work in this subject area.

THIS PAGE INTENTIONALLY LEFT BLANK

# II. CURRENT STATE OF VE INTERACTION

## A. VIRTUAL ENVIRONMENT INTERACTION

Interaction within a virtual environment can take on many forms enabling a wide range of techniques. Foley (1984) outlines the fundamentals of all user interaction, providing a template for analyzing user interaction in a VE. However, before VE interaction can be explored, it is important to understand some fundamental definitions.

- **2D Environment**: A 2D environment is one in which the location and definition of all objects are constrained to a single plane. Objects in a 2D environment have three degrees of freedom.
- **3D Environment**: A 3D environment is one in which the location and definition of all objects can be presented in up to three dimensions. It is important to note that although a desktop computer is normally used as a 2D environment, it becomes a window to a 3D environment when interaction with objects in the environment occurs in three dimensions. Objects in a 3D environment have six degrees of freedom.
- **Interaction**: Interaction is a mutual or reciprocal action or influence. (Webster's Revised Unabridged Dictionary, 1998).
- **Technique**: A technique is the systematic procedure by which a complex or scientific task is accomplished. (The American Heritage Dictionary of the English Language, 1996).
- **Computer Interaction**: Computer interaction is the set of actions taken by a user that result in reciprocal actions by a computer.
- **Interaction Task**: An interaction task is a fundamental task performed by the user that can not be further decomposed into sub-tasks. Execution of the task results in an appropriate reciprocal action by the computer. Interaction tasks can be performed using a variety of techniques. Each interaction task has specific requirements based on its parent application or parameters specified by the user.
- **Interaction Technique**: An interaction technique is a method used to accomplish an interaction task. Techniques involve a series of steps performed by the user in order to complete a task. Each technique has certain properties that define it. To further amplify the distinction between an interaction task and an interaction technique, consider that a positioning task could require a user to relocate an object within 3D space while a possible positioning technique might be capable of only performing two dimensional movement.

- **Interaction Device**: An interaction device is a piece of computer hardware, generally capable of a variety of interaction techniques, used to perform interaction tasks. An interaction device should not be confused with an interaction technique. For instance, in a Microsoft Windows environment, a file can often be opened using a variety of techniques with the same device. One can use a mouse to double click on a file, right click and select "Open" from a pull down menu, or drag and drop the file onto an executable program. All these interaction techniques achieve the same end-state, and they all use the same interaction device, the mouse.
- **Dimensional Ambiguity**: Dimensional ambiguity is defined as not possessing an inherent or generally accepted dimensionality. Tasks and techniques that are dimensionally ambiguous often *are* performed in more than one dimensionality (1D, 2D, or 3D), depending on the specific requirements of the task or the implementation design. However, just because a task or technique *can* be performed in more than one dimensionality does not mean that task or technique is dimensionally ambiguous.
- **Inherent Dimensionality**: An inherent dimensionality is a generally accepted dimensionality for a task or technique. For example, although a cursor could be positioned on a desktop by a series of two one-dimensional interaction techniques (e.g., slide-bars for x and y position), the task of positioning a cursor on a desktop is generally accepted to be a two-dimensional task. Thus, it would be correct to say that positioning a cursor on a desktop is an inherently 2D task.

## B. FUNDAMENTAL TYPES OF INTERACTION TASKS

There are six fundamental types of interaction tasks for all human-computer interaction (Foley, et al, 1984). They are:

- **Select**: Pick an object from a given set of objects
- **Position**: Move an object or icon from one location to another
- **Orient**: Change the heading, pitch, or roll of an object
- **Path**: Plot the position and/or orientation of an object over time
- **Quantify**: Associate a value or measurement with an object or concept
- **Text**: Enter a string of characters for use as a record or annotation

Foley defines these task types and associates representative examples of interaction techniques with each. Additionally, these task types can be represented either spatially or symbolically. It is the combination of the properties of these task types and

their typical representation that can lead to confusion and dimensional mismatches when implementations for each task are developed.

*Select* tasks require the user to make a selection from a group of objects in a given set. Objects can range from items in a list to 3 dimensional graphical representations of real world objects. Typical interaction techniques associated with *Select* tasks include menu selection with a pointing device, object picking with a pointing device, keyboard input of alphanumeric identifiers or function keys, and voice input. It is necessary to point out that although the aforementioned techniques span the range of spatial dimensions from 1D to 3D, the *Select* task itself is inherently 1D. The *Position* task that is performed prior to a *Select* task in order to locate a cursor or a pointer over the desired object is a distinct task and should not be combined with or confused with the actual selection of the object. It is also interesting to note that the dimensionality of the *Position* task performed prior to the *Select* task generally coincides with the dimensionality of the object being positioned. Because the object set on which a *Select* task can be performed is neither inherently 2D nor 3D and the typical techniques employed to perform a *Select* task extend across the range of spatial dimensions, the dimensional sensitivity of the task can not be limited to a specific dimensionality. Therefore, *Select* tasks can be dimensionally ambiguous and present a dilemma to the VE designer when selecting appropriate interaction techniques for a given application.

*Position* tasks can also be dimensionally ambiguous. To perform these types of tasks, the user must indicate a location on the interactive display, usually identifying where an object is to be placed within the environment. In this case, objects can include

icons, text, various 2D/3D graphics, or the user's viewpoint. Interaction techniques used to perform *Position* tasks are also very similar to those associated with *Select* tasks. Typical techniques are positioning of a cursor icon on a display using a mouse, joystick, or other pointing device, moving files or folders from one directory location to another, entering positioning coordinates via a keyboard or number pad, and moving a slide bar laterally or vertically. Note that the *Position* task does not include the actions performed to select the objects, such as files, folders, slide bars, etc., to be moved, but only their actual movement from one location to another. Since *Position* tasks can occur in one (slide bar), two (cursor on a desktop), or three (graphical object in a VE) dimensions, *Position* tasks can be dimensionally ambiguous. This can often lead to dimensionality conflicts between task requirements and available interaction techniques.

*Orient* task characteristics are similar to those of *Position* tasks. An *Orient* task requires the user to orient an object in 2D or 3D space. Objects affected by an *Orient* task are the same as those affected by a *Position* task. It is interesting to note, however, that while the number of orientation angles that can be manipulated to change the 3D orientation of an object is three, only one angle can be affected when changing the 2D orientation of an object. This distinction is mirrored when *Position* and *Orient* tasks are combined to reflect the degrees of freedom of an object. An object whose 3D position and orientation can be manipulated is said to have six degrees of freedom, or translation in the X, Y, and Z planes and rotation about the object's X, Y, and Z-axis. An object whose 2D position and orientation are the only spatial properties that can be adjusted is described as having three degrees of freedom, or translation along 2 of 3 spatial axes and

rotation about the 3rd. Representative interaction techniques for *Orient* tasks include control of orientation angles using a mouse, joystick, or other pointing device and keyboard entry of angular changes. The clear difference between the nature of the *Orient* task in a 2D versus a 3D environment is also reflected in an *Orient* task's dimensional sensitivity. Orientation of an object within a 3D environment is dimensionally ambiguous, since the object's orientation may be affected in one, two, or three dimensions, depending on the specific application requirements. However, orientation of an object in a 2D environment is clearly constrained to rotation about a single axis. This constraint can lead to a dimensionality conflict when a user task requires greater degrees of freedom.

A *Path* task is defined as a series of position and orientation changes occurring over time. Even though a *Path* task contains other primitive *Position* and/or *Orient* tasks, it is perceived differently by the user because of the introduction of the element of time. While performing a *Position* or an *Orient* task, the user is concerned solely with the end state of that task, whereas their focus during the performance of a *Path* task is on a series of positions and orientations and the order in which those events occur. The objects on which a *Path* task can be performed and the interaction techniques that are typically used are the same as those associated with a *Position* or an *Orient* task. Thus, a *Path* task can also be dimensionally ambiguous.

A *Quantify* task has no inherent dimensionality. Rather, it is a measurement, such as the height or length of time. Although the object whose properties are being quantified may have a very clear dimensionality, its quantified dimensional measurements have no

inherent dimensionality themselves. Thus, it also follows that a *Quantify* task is not constrained to a physical or virtual object, but can be applied toward a concept or event as well. Typical techniques used to complete these types of tasks include entering values using a keyboard or assigning values by positioning a slide-bar.

Unlike any of the previously mentioned tasks, the *Text* task is one whose presentation is entirely symbolic rather than spatial. Written languages generally have been represented by some form of two-dimensional symbology, most often classified as text. Text therefore has an inherent dimensionality. Although text may be represented using either two or three-dimensional characters, the *Text* task, as history and common use has shown, is inherently two-dimensional. *Text* tasks require the user to enter a string of alphanumeric characters that usually have semantic content associated with a language. This task should not be confused with techniques used to perform other types of tasks. A simple way to distinguish between a *Text* task and a technique that involves textual input is that a string entered in the performance of a *Text* task is stored on the computer as data for later use or viewing, and is not used as a command or converted to a value, position, or orientation for the purpose of accomplishing one of the other task types. Typical interaction techniques used to perform a *Text* task are alphanumeric keyboard entry, handwriting recognition, speech recognition, and character selection from a menu.

It is this last task type where the most problems occur with regard to mismatching the dimensionality requirements of a task and the technique used to perform it. Unfortunately, "current alphanumeric input techniques for the virtual world (which we use for precise interaction in the computer world) are ineffective" (Mine, 1995), and

therefore dimensionality mismatches are rather common. A further examination of existing interaction techniques in virtual environments makes it abundantly clear that while *Text* tasks are not the only task type for which dimensionality conflicts occur, they are the predominant source of such problems. A synopsis of task types and their dimensionality properties appears in Table 2.1.

| TASK | SPATIAL | SYMBOLIC | DIMENSIONAL AMBIGUITY | INHERENT DIMENSIONALITY |
|---|---|---|---|---|
| SELECT | **X** | | | **1D*** |
| POSITION | **X** | | **X** | |
| ORIENT | **X** | | **X** | |
| PATH | **X** | | **X** | |
| QUANTIFY | * | ** | | **NONE** |
| TEXT | | **X** | | **2D** |

* Although this task is inherently 1D, the dimensionality requirements for this task type are generally associated with its accompanying *Position* or *Orient* task and the dimensionality of the object being selected.
** As described above, *Quantify* tasks are not necessarily spatial, nor are they symbolic.

Table 2.1. Tasks Types and Associated Properties.

## C. EXISTING VIRTUAL ENVIRONMENT INTERACTION TECHNIQUES

Given the task types described above, it is important to now examine existing applications and techniques in order to more clearly understand the problem that exists in virtual environments when tasks and techniques are mismatched with regard to their dimensional requirements. As it would be nearly impossible to examine all techniques that currently exist, an attempt will be made to look at a representative sample of those

that are being used both in VE-based training and in VE research, focusing first on interaction techniques used in the completion of 3D tasks and then examining existing techniques for performing inherently 2D tasks.

### 1. 3D Interaction Techniques

Virtual Environments provide the user with a graphical representation of a three-dimensional environment. Therefore, one must have techniques available that enable interaction with that environment. Several techniques have been developed to enable such interaction. Examples are described in detail below.

#### *a) Two-handed Direct Manipulation*

This interaction technique is used in a wide variety of VE applications, including those using a CAVE Automated Virtual Environment (CAVE), a Head-Mounted Display (HMD), or a Virtual Workbench. The interaction device used to

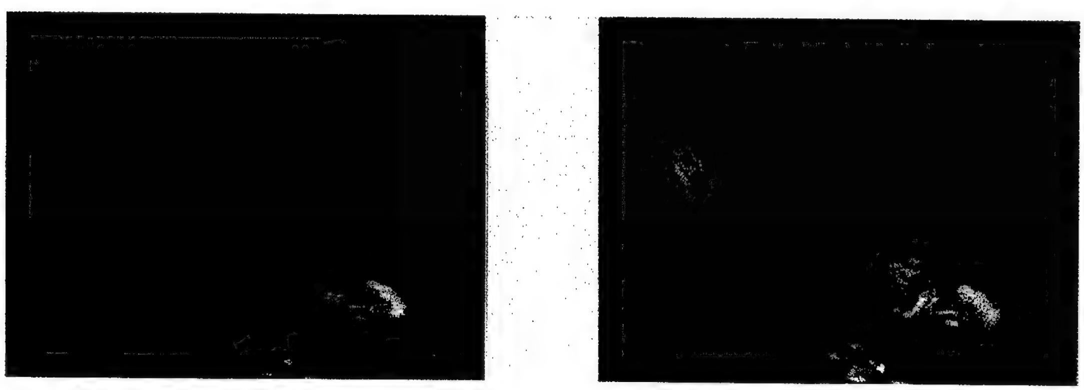


Figure 2.1. Pinch Glove and 6 DOF Stylus Interaction (Cutler, et al., 1997).

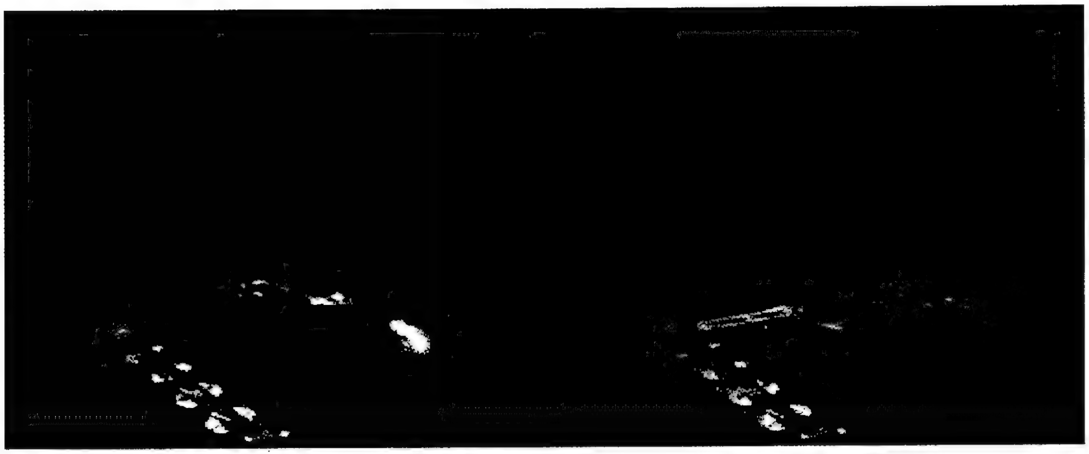


Figure 2.2. Two Pinch Glove Interaction (Cutler, et al., 1997).

perform this technique is the data glove, also referred to as a pinch glove. Pinch gloves communicate hand locations to the Virtual Environment using tracking technology and also communicate when fingertips and thumbs are touching each other via sensors located a the tip of each finger. One system in particular, developed by the Graphics Department at Stanford University (Cutler, et al., 1997), allows users to naturally manipulate virtual 3D models with both hands on the Responsive Workbench, a tabletop VE device. Users manipulate the objects using either a data glove and a tracked stylus or two data gloves

Figure 2.3. Selecting a Manipulation Technique from the Tray (Cutler, et al. 1997).

(Figures 2.1 and 2.2). Users choose manipulation techniques from a menu tray presented on the front edge of the workbench, or by gestures performed with the pinch gloves (Figure 2.3). Using Foley's six task types, this technique is used to perform *Select, Position,* and *Orientation* tasks. The *Select* task is accomplished by first performing a 3D position task, locating the data gloves so that manipulation techniques can be selected from the tray at the front of the workbench. Upon selecting the manipulation technique from the tray, the user is able to perform *Position* and *Orient* tasks on graphical images on the workbench by pinching with the gloves to grasp objects and then moving and orienting the objects just as one would if holding a real object. In this case, the dimensionality of the interaction tasks performed matches the dimensionality of the interaction technique used to perform them. The result is a natural interaction that is easily accomplished by the user.

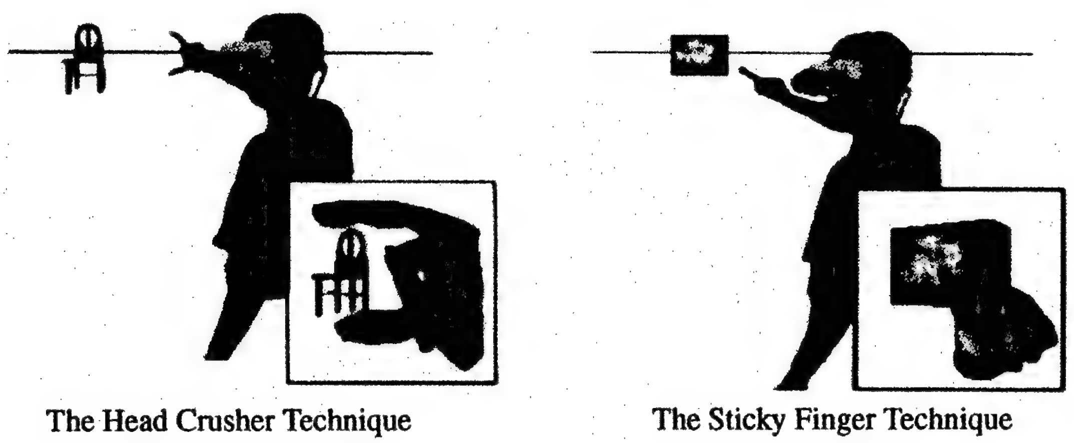



Figure 2.4. Head Crusher and Sticky Finger Techniques (Pierce, et al., 1997).

### b) *Image Plane Interaction Techniques*

Image plane interaction techniques were developed in a collaborative project between researchers at the University of Virginia, Brown University, and the University of North Carolina (Pierce, et al., 1997). The interaction devices used to accomplish these techniques were a head-tracked HMD and data gloves. The first of these techniques, the Head Crusher technique (Figure 2.4), enables the user to grasp an object in the scene by placing his finger and thumb above and below (respectively) the 3D object to be manipulated as it appears in the 2D image plane. The object can then be manipulated by actions performed with both hands. The Sticky Finger technique uses an easier gesture to select objects (Figure 2.4). The user places an index finger over the object to be selected, as it appears in the 2D image plane. The object is selected by casting a ray into the scene from the user's eye location through the tip of the index finger. Objects intersecting that ray are selected. The object is then manipulated by

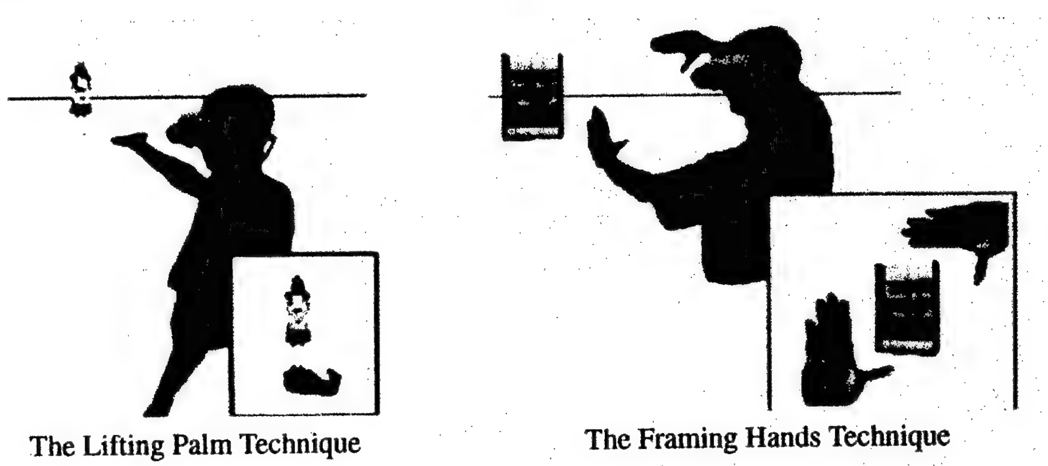



Figure 2.5. Lifting Palm and Framing Hands Techniques (Pierce, et al., 1997).

actions performed with both hands. The Lifting Palm technique requires the user to extend a hand so that the palm is facing up (Figure 2.5). The system then computes an offset to determine a position that is slightly above the palm. A pick ray is sent out from the eye position through the offset position. Objects intersecting that ray are selected and then manipulated by movement made with the lifting palm. The final image plane technique, the Framing Hands technique, enables the user to use both hands to frame the 3D object as it appears in the 2D image plane. The system determines the mid-point between the two hands and projects a pick ray from the eye location through that mid-point. Objects intersecting that ray are selected and can be manipulated by movements made with one of the two hands. All four of these image plane techniques are three-dimensional techniques and are used to perform *Select*, *Position*, and *Orient* tasks. In the applications described by Pierce, et al., these techniques are applied to 3D objects in a scene, thereby matching the dimensionality of the technique to the dimensionality of the task. The result is a technique that allows the user to select and manipulate 3D objects in the scene easily.

#### *c) Arm Extension Technique*

The arm extension technique was developed by Poupyrev, Billinghurst, et al., (1996) in a collaborative effort between researchers at Hiroshima University and the University of Washington. This technique, also referred to as the "go-go" technique, enables the user to grab and manipulate remote virtual objects in an immersive virtual environment. In this technique the user's virtual arm is made to grow at a non-linear rate proportional to the extent that the user's physical arm is moved away from the body. This

enables users to grab objects at a finite distance. However, because of the non-linear growth rate, hand position is difficult to control. Manipulation of objects, once grabbed, is very intuitive; the object moves in position and orientation relative to the movement of the physical (and therefore virtual) hand.

The "stretch go-go" technique (Bowman and Hodges, 1997) additionally allows the user to grab objects at potentially infinite distances. This process is controlled by the extent to which the physical arm is moved away from the body. When the physical arm is fully extended, the virtual hand arm extends at a linear rate. When the physical arm is pulled in close to the body, the arm retracts at a linear rate. The location of the

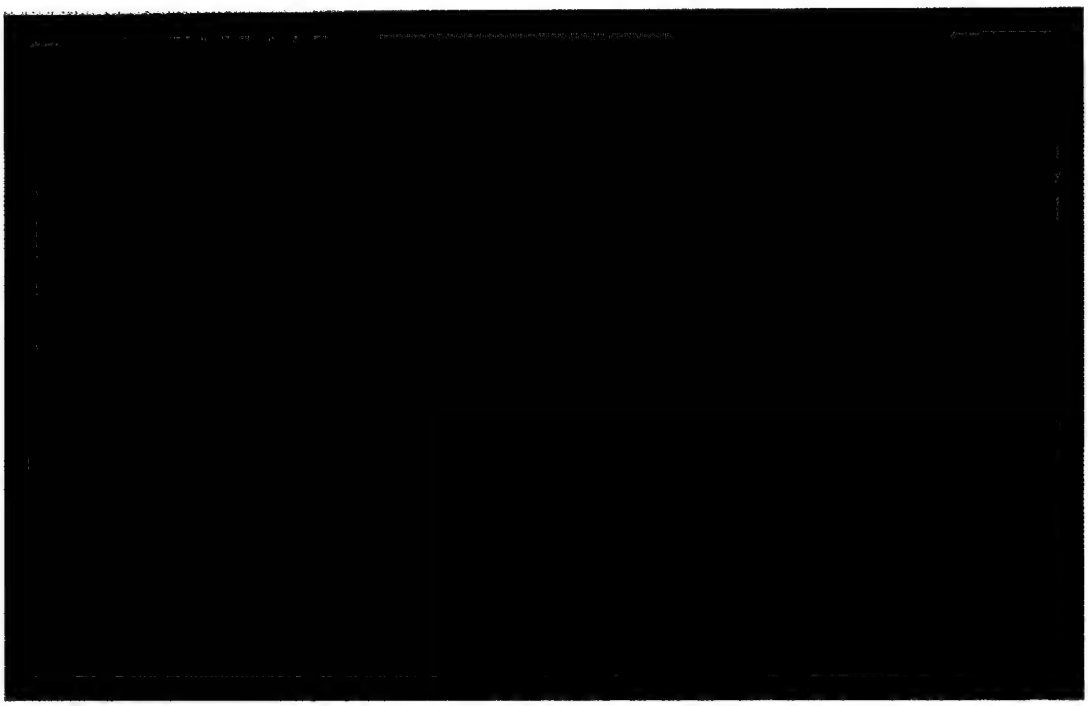


Figure 2.6. Stretch Go-Go Technique.

physical arm, and thus the rate of extension or retraction, is displayed on a slide bar to the right of the scene (Figure 2.6). An obvious human-factors drawback to this technique is the arm fatigue that occurs from having to maintain the physical arm at a partially extended position. This consequently makes arm length difficult to control.

Another modification of the go-go technique is the "indirect stretching" method. This method attempts to resolve the human factors issue mentioned above by replacing arm extension / retraction with mouse interaction. The virtual arm can therefore potentially be extended to an infinite distance. All manipulation occurs just as it would in basic arm extension.

All of these image plane techniques are used to perform *Select*, *Position*, and *Orient* tasks on 3D objects, thus matching the dimensionality of the technique to the dimensionality of the task. Although none of these techniques provide a method for performing inherently 2D tasks such as a *Text* task, they do provide the user with an effective way of interacting with a 3D scene and performing 3D tasks. The ability to interact with objects located at a distance, however, is limited and somewhat difficult to accomplish.

***d) Ray-Casting***

The ray-casting technique enables the user to select objects in the virtual environment by shooting a virtual ray from the hand into the scene along the direction in which the hand is pointed (Mine, 1995). Objects intersecting that beam are then selected and can be manipulated (Figure 2.7). Manipulation, however, is extremely difficult, as the object is not hand-centered. Instead the user encounters a "lever-arm" problem, in

which the selected object is in essence attached to the end of a long lever arm. This makes controlling the distance of the object from the user impossible and makes all other forms of manipulation extremely difficult.

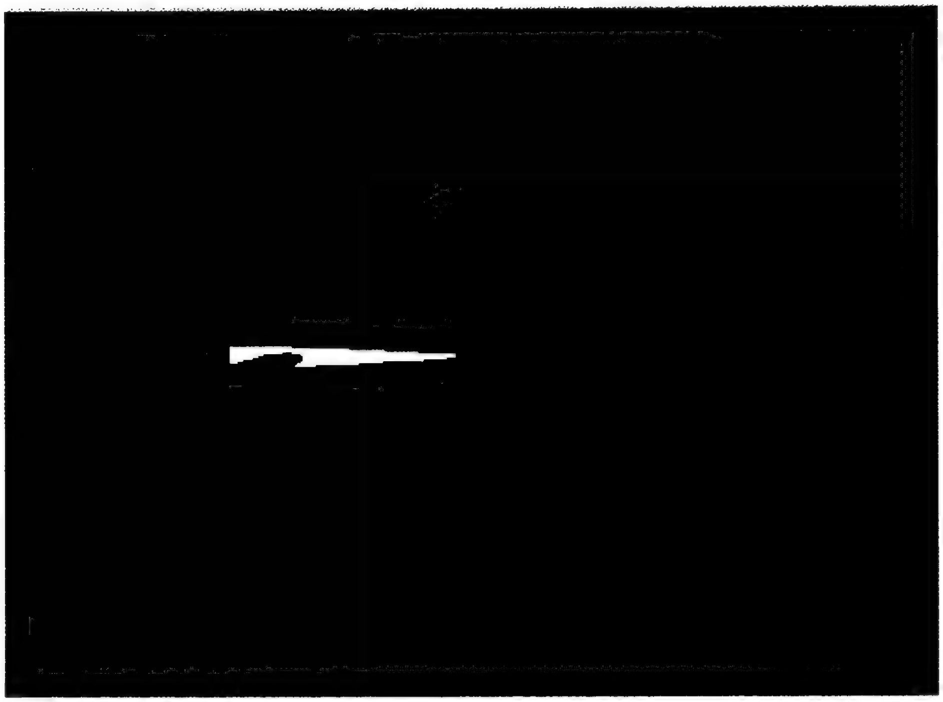


Figure 2.7. Ray-Casting Technique (Mine, 1995).

A modified form of ray-casting developed in 1997 added a reel-in feature (Bowman and Hodges, 1997). This modified form associates a technique similar to one used in the indirect stretching method. The user is able to control the distance of a selected object by using mouse buttons to "reel" the object in or out. Other position and orientation tasks are still quite difficult, however, as the "lever-arm" problem is not alleviated.

Ray-casting, therefore, solves some of the problems that exist with the arm extension techniques, enabling the user to more easily perform *Select* and *Position* tasks on distant objects in the scene. The ray-casting technique does have a clear disadvantage when the user is required to perform *Orient* tasks, as the "lever-arm" problem makes orientation of all but the closest of objects virtually impossible. Additionally, none of the ray-casting techniques provide a means for performing *Text* tasks.

***e) HOMER***

Bowman and Hodges (1997) developed a technique that combined the strengths of the arm extension and ray-casting techniques called **H**and-centered **O**bject **M**anipulation **E**xtending **R**ay-casting (HOMER). This method allows users to select an object in the scene using a light ray, as was the case in ray-casting. Once selected, the object becomes hand-centered, enabling the ease of manipulation found in the arm extension techniques. Positioning of the object is coupled to the relative distance of the users' physical hand from their body. Moving their hand half-way between their body and full extension moves the object half-way between the user and the object's initial position. Most distances can be obtained with practice.

A variation of the HOMER technique, called indirect HOMER, provides users with greater precision and unbounded reach. Distance of the object from users is controlled using mouse buttons, and manipulation occurs as it does in direct HOMER. Both HOMER techniques enable users to perform *Select*, *Position*, and *Orient* tasks on 3D objects in the VE. By implementing the best features of arm extension and ray-casting, it provides users with a flexible and very capable interaction technique for

performing 3D tasks. However, similar to the arm extension and ray-casting techniques mentioned above, HOMER does not provide users with a means for executing any 2D tasks.

#### *f) Two Pointer Input*

Zeleznik, Forsberg, and Strauss (1997) developed a technique for using two pointing devices as input devices for 3D interaction in 3D desktop applications, thereby enabling the user to perform two-handed interaction with objects in the environment. The technique involves the use of a mouse in the non-dominant hand and a stylus in the dominant hand. Both the mouse and the stylus have buttons that are used by the system to interpret the actions performed with the mouse and stylus. Combinations of button pushes and hand movements with both pointing devices enable the user to build and manipulate objects in the scene. One approach to two cursor input involves the use of absolute input devices, such as a puck and a mouse on a tablet. This approach presents some physical problems for the user. The user's hands sometimes interfere with each other on the tablet due to either a requirement for them to work in close proximity to one another or because the task or the implementation may require the user to reach one hand across the other. A second approach implements relative input devices such as two mice. This approach requires greater dexterity on the part of the user and can thus be a more difficult technique to use in performing *Position* and *Orient* tasks. Both two-pointer approaches are better suited for use with a virtual workbench than with more immersive hardware such as a CAVE or HMD. Despite the haptic feedback provided by the tablet and the appearance that the use of the stylus should provide a simple means for

performing *Text* tasks, the two pointer input does not provide the user with any way to perform such tasks.

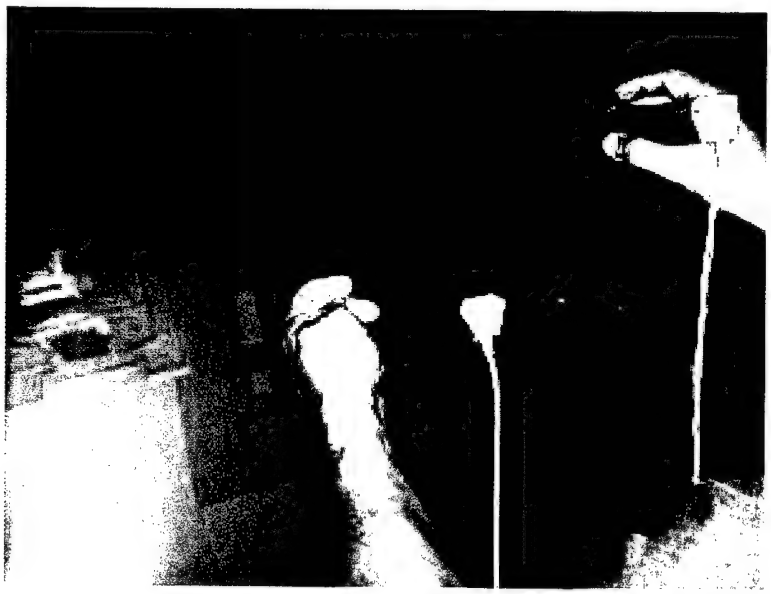


Figure 2.8. Transparent Props (Schmalstieg, et al., 1999).

### *g) Transparent Props*

This technique is used with the virtual workbench and was developed by Schmalstieg, Encarnação, and Szalavári at the Vienna University of Technology (1999). It is based on transparent props that are augmented with 3D graphics from the virtual workbench display and allows for a variety of interaction techniques. Transparent props require two-handed interaction and introduce the 2D paradigm into the 3D environment by providing the user with a transparent pad and a tracked hand-held pen with which to select and manipulate objects in the scene (Figure 2.8). The two props also combine several metaphors. The pad can be used as an object palette to carry tools and controls that can be selected using the pen. It can also be used to take a "snapshot" of a portion of

the 3D scene on the workbench, enabling the user to replicate and manipulate objects on the workbench (Figure 2.9). It is important to note that although the 3D objects on the virtual workbench are displayed on the 2D surface of the pad when using this technique, the dimensionality of the device should not be confused with the dimensionality of the

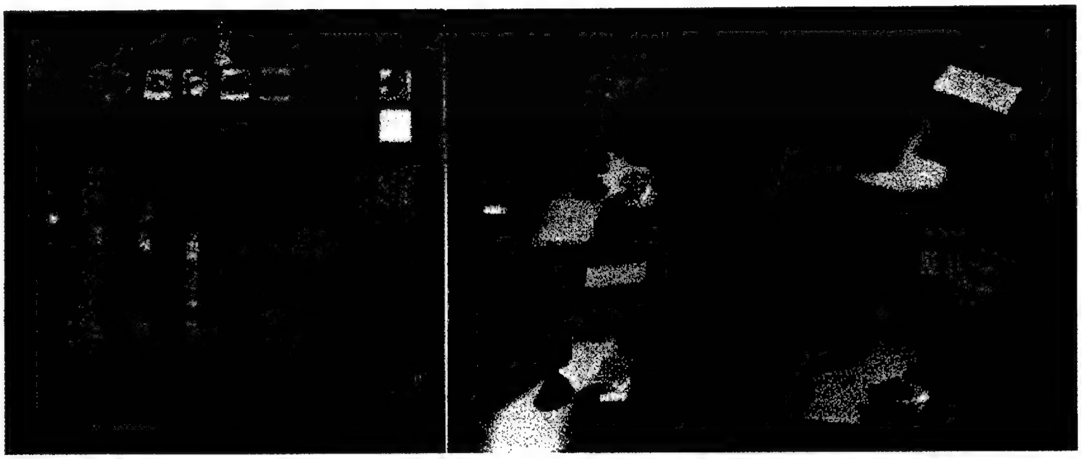


Figure 2.9. Transparent Props as a Palette and as a Snapshot Tool (Schmalstieg, et al., 1999).

interaction technique. The techniques that incorporate the snapshot and volumetric manipulation enabled by these devices are 3D interaction techniques. All these interaction techniques enable the user to perform *Select*, *Position*, and *Orient* tasks on 3D objects on the virtual workbench, thereby matching the dimensionality of the task to the dimensionality of the technique. However, none of them provide a method for performing 2D *Text* tasks despite the presence of devices that are typically associated with 2D interaction techniques.

#### *h) CHIMP*

Mine (1996) developed the Chapel Hill Immersive Modeling Program (CHIMP) at the University of North Carolina. CHIMP provides a variety of ways to select and manipulate objects in the virtual environment. The user can perform one or two-handed interaction using two separate bats with 6 degrees-of-freedom (DOF), one for each hand. A bat is a hand-held input device that contains a tracking sensor to detect the location and orientation of the user's hands. Each bat also has several buttons that are

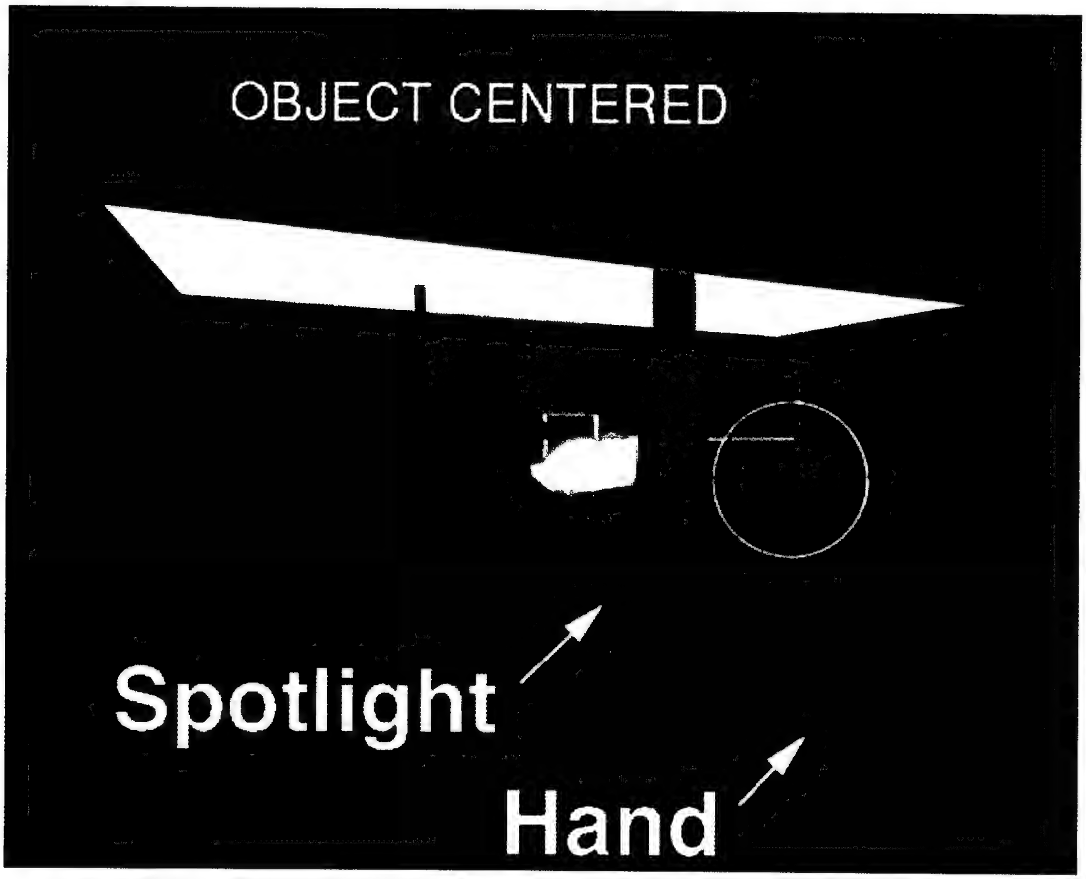



Figure 2.10. Spotlight Selection Technique in CHIMP (Mine, 1996).

used to allow various kinds of manipulation with each hand. Similar to ray-casting, CHIMP uses a spotlight that is projected from the virtual hand location to select objects in the scene (Figure 2.10). The spotlight is preferred over ray-casting in CHIMP because it does not require as high a degree of precision by the user, thereby facilitating selection of small targets at long range. There are also numerous pop-up menus, called look-at menus, located throughout the scene. Some are tied to objects; others are for manipulation and configuration of the scene in general. Light-colored circles indicate the location of the menus in the scene. Placing the spotlight within the circle and selecting

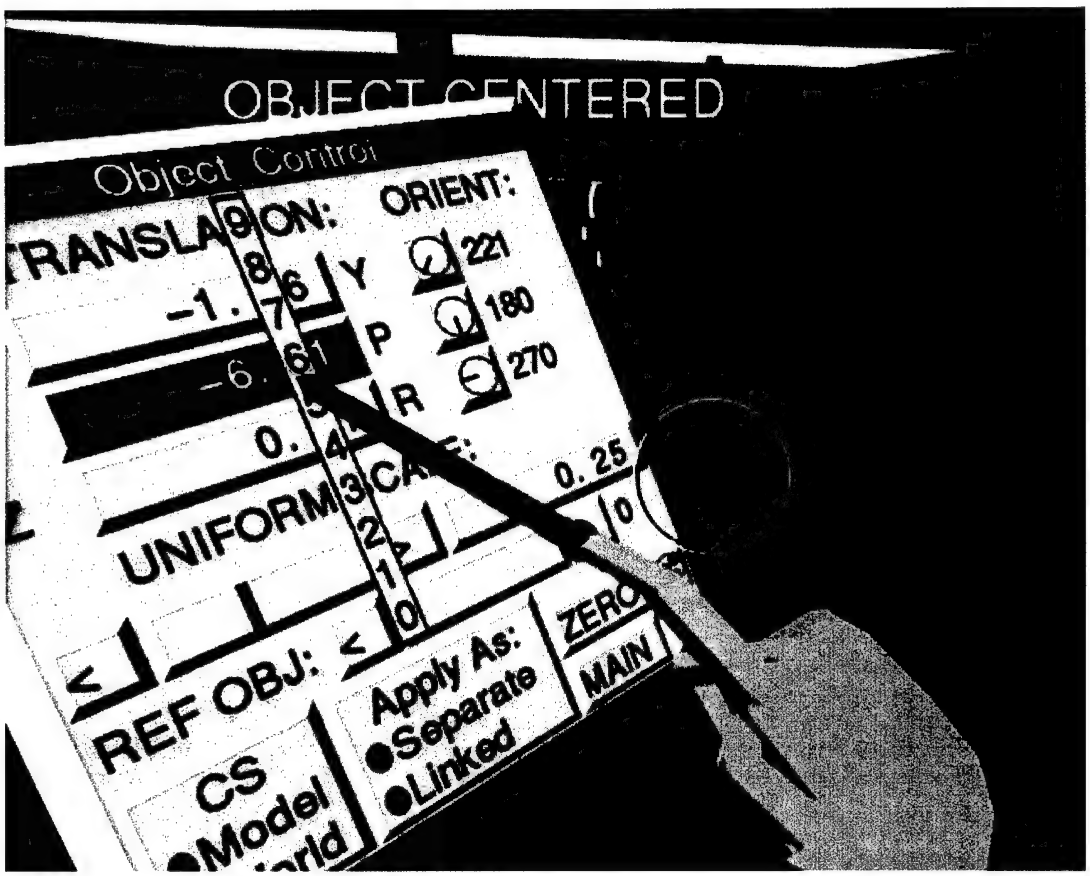



Figure 2.11. Number Entry with CHIMP (Mine, 1996).

brings up the menu. The environment also contains control panels that are the equivalent of dialog boxes in a Window, Icon, Menu, Pointer (WIMP) interface. When control panels are active, they are attached to the user's left hand, presenting a 2D interface in the 3D environment. However, users will only notice these control panels if their left hand is in a location where it can be easily viewed. Users then use a bat in the right hand to select items on the menu and also to perform *Text* tasks such as number entry (Figure 2.11). The techniques used in CHIMP provide a wider range of capabilities than any of the previously mentioned techniques, because CHIMP includes the ability to not only perform 3D *Select*, *Position*, and *Orient* tasks, but also the ability to perform *Text* tasks. However, the technique used to perform the *Text* task is a 3D technique. For numeric input, the user wields a 6 DOF bat, held in the right hand, to point at a virtual menu that is held in the left hand. The user then selects each digit of the numeric value from a pull-down list. The dimensionality mismatch between the 2D *Text* input task and the 3D interaction technique makes performing the task awkward and difficult for the user.

### 2. 2D Interaction Techniques

The interaction techniques discussed in the previous sections were used primarily for 3D interaction in virtual environments and on virtual workbenches. CHIMP was the only one that provided a means for accomplishing 2D *Text* tasks, although the majority of the interaction techniques available in CHIMP are intended for use in accomplishing 3D tasks. The inability to perform 2D *Text* tasks in a VE is a major shortcoming in current VE applications. The reason that many current attempts to introduce the 2D interaction

paradigm into 3D VE applications fail is that many of them mismatch the dimensionality requirements of the task and the dimensionality of the technique. The following sections highlight examples of current 2D interaction techniques used in VE applications.

#### *a) Virtual Menus*

Virtual menus are an attempt to introduce a standard WIMP 2D interface into 3D virtual environments. Virtual menus are generally presented in one of two configurations. One configuration presents the virtual menu to the use by floating it in 3D space, providing the user with only visual stimuli and no tactile or haptic feedback. This configuration clearly requires the user to perform a 3D interaction, despite the representation of the menu as a two dimensional object in the VE. Typical techniques used to select an object from a floating menu include ray-casting (Bowman and Hodges, 1997), grasping with a data glove (Cutler, et al., 1997), or using a spotlight (Mine, 1996). The problem this configuration presents, besides the lack of haptic feedback, is that it requires the use of 3D interaction techniques to perform 2D *Select* and *Position* tasks, thereby essentially turning those 2D tasks into more complicated 3D tasks. Additionally, when the VE application requires the user to perform a *Text* task using these techniques, the same dimensionality mismatch occurs.

The second configuration presents the virtual menu on a hand-held tablet or paddle, used as a prop in conjunction with an HMD. Lindeman, et al. (1999) have shown that users perform 2D tasks in virtual environments faster and with fewer errors when they are provided with passive haptic feedback in the accomplishment of the task. In his experiment, Lindeman presented the subjects with hand-held and world-fixed 2D

displays. The subjects were required to perform a *Select* and a *Position* task using both display types with and without passive haptic feedback. Although the results showed the presence of passive haptic feedback resulted in faster task performance with a greater degree of accuracy, the results also suggest another finding.

The technique used to perform the *Position* task when no passive haptic feedback was available was three-dimensional, despite the dimensionality requirements of the task being two-dimensional. This dimensionality mismatch may account for the significant difference in the time required to perform the task and the number of errors that resulted. Subjects required almost twice as much time and committed almost twice as many errors when no passive haptic feedback was available. When passive haptic feedback was provided, the interaction technique constrained the users' actions to a single plane by providing either a paddle or tablet in the case of the hand-held display, and a rigid Styrofoam box in the case of a world-fixed display. Thus the dimensionality of the interaction technique matched the dimensionality requirements of the task.

The *Select* task resulted in only slight differences between performance with and without passive haptic feedback. The *Select* task does not demonstrate the same dimensionality mismatch dilemma that occurred with the *Position* task, because the *Select* task type is dimensionally ambiguous. In this case, the dimensionality of the task matched the dimensionality of the technique used to perform the task, since the dimensionality requirements of a *Select* task are generally closely associated with the dimensionality of the object being selected. When the passive haptic feedback was available, the display was presented on a 2D surface and the associated interaction

technique was constrained to the dimensionality of that surface. When no passive haptic feedback was available, the display was presented as an object in the 3D scene. Thus the *Select* task associated with the display was treated as a 3D *Select* task, since the display object was a 3D object in the scene. As a result, the interaction techniques used to select an item on the display matched the dimensionality of the task.


Figure 2.12. Virtual Notepad (Poupyrev and Tomokazu, 1998).

***b) Virtual Notepad***

The Virtual Notepad was created in a collaborative effort between Poupyrev and Tomokazu at Hiroshima University and Weghorst at the University of Washington (1998). This research also introduces the 2D interface into an immersive virtual environment by providing the user with a pressure sensitive pad and a pen and is designed specifically for the performance of *Text* tasks in a VE (Figure 2.12). Given that *Text* tasks are inherently 2D, this technique is a refreshing change to other proposed VE

interaction techniques for performing such tasks. The user is provided with a small tracked tablet that becomes visible only when the pen touches it. The user is able to write notes, erase mistakes, "tear" notes off the pad and place them in the environment, and flip through the Virtual Notepad to look at other notes that were written earlier. This technique is intuitive and clearly matches the dimensionality of the task to the dimensionality of the technique; however, it is exclusively 2D, providing no means for performing 3D tasks.

### *c) Hand-held Computers in Virtual Environments*

Watsen, Darken, and Capps, from the Naval Postgraduate School, developed the concept of using a hand-held computer, such as a PalmPilot or other personal digital assistants (PDAs) in a virtual environment (1995). This concept evolved from Wloka and Greenfield's work with a Virtual Tricorder (1995). Their desire was to bring a device capable of normal 2D interaction into the 3D environment without sacrificing the advantages and functionality of the VE. The attempt demonstrates some promise of success, as users are able to use the PDA to perform 2D interaction by using a 2D interaction technique on a 2D device, without sacrificing display space, as often occurs with techniques associated with HMDs. Though the use of a PDA enables a dimensionality match between task and technique for 2D tasks, that is not the case for 3D tasks performed in the VE. In the test implementation, the PDA is used to navigate through 3D space and perform 3D *Position* and *Orient* tasks on objects in the scene. This dimensionality mismatch occurs when the user is required to perform 3D tasks using 2D techniques, thereby diminishing performance.

#### *d) Desktop Virtual Environments*

Virtual Reality Modeling Language (VRML) is an example of a language used to create VE applications for the desktop. These applications provide another interesting dimensionality. The primary interaction device used in conjunction with VRML applications is the 2D mouse, a device capable of only 2D interaction techniques. However, the majority of the tasks performed in VRML applications are 3D — thus presenting another dimensionality conflict between task requirements and available interaction techniques. Computer users, in general, have become quite adept in using the mouse to perform tasks of many dimensionality requirements. The most obvious reason for this adaptation with regard to 3D environments is that immersive VE hardware is not widely available to the general user. In an effort to make VE technology available to a larger audience, many VE applications have been modified to work in a desktop environment. They provide a window to a VE, rather than the fully immersive experience that becomes available with the introduction of HMD and CAVE-type technologies. As VE technology matures and becomes more widely available, the current 2D interaction techniques associated with desktop VE will no longer be a satisfactory means for performing 3D tasks. The user will require another means of interacting with the environment — one that matches the dimensionality requirement of both technique and task.

### 3. Summary

Current VE interaction techniques utilize a wide range of devices to perform both 2D and 3D tasks. Table 2.2 provides a quick synopsis of some of the major

techniques that are currently available and their ability to perform various task types without mismatching dimensionalities. Note that the *Path* task type is not included because it was not discussed in association with any of the techniques that were examined, and it is generally associated with the development, rather than the use of an application. The *Quantify* task type is not included because it has no inherent dimensionality.

| Interaction Technique | Select Task | Position Task | Orient Task | Text Task |
|---|---|---|---|---|
| 1. 3D Mouse | Yes | Yes | Yes | No |
| 2. Two Handed Direct Manip. | Yes | Yes | Yes | No |
| 3. Image Plane | Yes | Yes | Yes | No |
| 4. Arm Extension | Yes | Yes | Yes | No |
| 5. Ray-casting | Yes | Yes | No | No |
| 6. HOMER | Yes | Yes | Yes | No |
| 7. Two Pointer | Yes | Yes | Yes | No |
| 8. Transparent Props | Yes | Yes | Yes | No |
| 9. CHIMP | Yes | Yes | Yes | No |
| 10. Virtual Menus | Yes | No* | No | No |
| 11. Virtual Notepad | No | No | No | Yes |
| 12. Hand-held computer | No | No | No | Yes |
| 13. VRML | No | No | No | Yes |

* The mismatch of dimensionalities in this case results from the use of 2D techniques on the virtual menu, such as slide bars or dials, not the technique used to interact with the virtual menu.

Table 2.2. Summary of 2D and 3D Interaction Techniques.

Clearly no single technique allows the user to accomplish all task types without incurring a dimensionality mismatch. In order to provide the user with a means of accomplishing tasks requiring both 2D and 3D interaction techniques, a different approach must be used in developing VE applications.

# III. APPROACH

## A. INTRODUCTION

This chapter describes an approach to designing virtual environment applications. The approach provides a framework for analyzing a VE application, considering the dimensionality requirements of all the tasks intended to be performed in the application as well as the techniques and devices available to accomplish those tasks.

## B. IMPACT OF DIMENSIONALITY ON VE DESIGN

> [Virtual Reality (VR)] will remain inferior to the desktop as a serious work environment until users of VR can access the same data as available on the desktop. ...Unless users have access to all the data they need to make intelligent decisions, VR interfaces will only provide a partial solution, one that may in the end hamper rather than enhance users' ability to perform work (Angus, 1995).

Ineffective 2D interaction techniques in VE applications hinder users' ability to access the same data normally available on a desktop computer. In order to solve this problem, not only must effective 2D interaction techniques be developed, but the approach to VE design must also change.

Schlager explored the issues surrounding the design of virtual environment training systems (1994). He felt it was critical for developers to determine an effective means for specifying system requirements for VE applications as well as considering what task characteristics indicate a VE is needed for training and how to determine the cost-effectiveness of a VE system. In order to determine what hardware was required for

a VE training environment, Schlager proposed conducting task analyses. Then, using requirement matrices based on task constraints, training impact, and learning outcomes, it would be possible to select the component technologies required to effectively use the VE application. A similar approach can be used for determining interaction device requirements when designing a virtual environment. Figure 3.1 provides a picture of an approach to VE application design that considers the dimensionality requirements of tasks and capabilities of techniques.

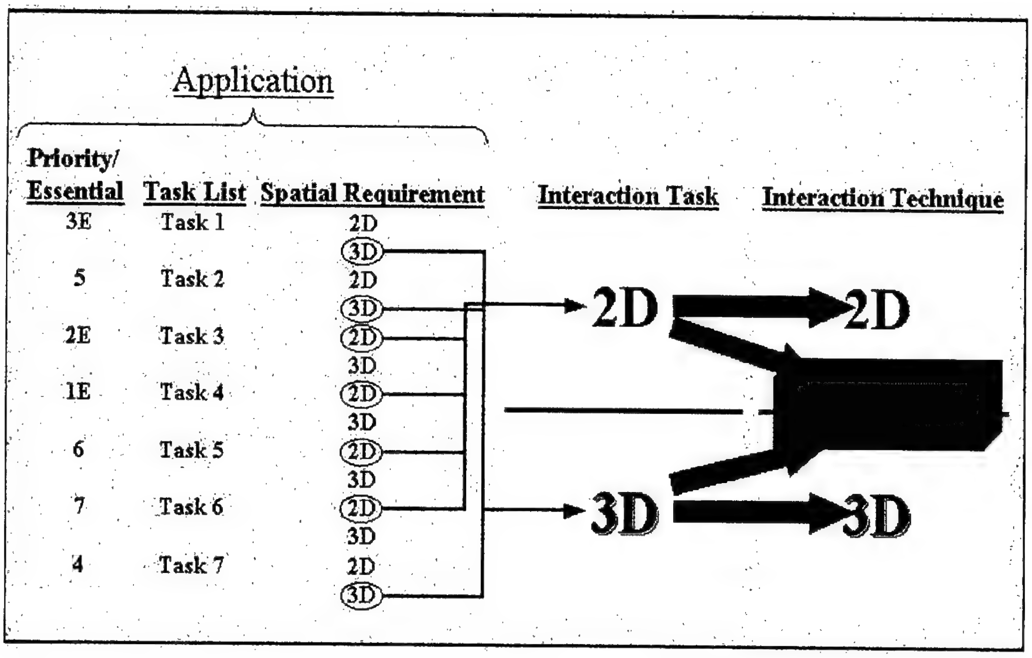



Figure 3.1. Approach to VE Application Design.

### 1. Task Decomposition

In order to be able to understand the hardware requirements for a virtual environment application, it is essential to examine the application and identify the fundamental tasks that must be performed. Foley's classification provides the necessary

basis for categorizing each task that can be performed in any given application. For instance, a VE application designed for engineering design review would likely require a variety of task types to be performed. In that example, the engineer might view new engine components, determine how to best position the components in multiple engine types, and record observations and recommendations regarding the new components. The engineer would need to perform *Select* tasks in order to pick components to examine and engines with which to associate the new components. *Position* and *Orient* tasks would be necessary to enable the engineer to properly place the new components in various engine types and also to position a pointer for selecting objects, if that were the implementation that was chosen. *Path* tasks might be required if the component were dynamic and needed to change position or orientation over time. *Quantify* tasks, such as dimension measurements and performance ratings, might need to be recorded. *Text* tasks would be necessary for recording the engineer's comments and recommendations. Since many applications do not contain all six task types, not all of the examples mentioned above would necessarily be required in the VE application design.

### 2. Dimensionality Categorization

Once all the tasks associated with a VE application have been identified and classified as one of the six fundamental task types, it is then necessary to examine each task to determine its dimensionality requirements. As mentioned previously in Chapter Two, certain task types have an inherent dimensionality, some have no inherent dimensionality, and still others are dimensionally ambiguous. It is important, therefore,

to identify the dimensionality requirements of each task, so that interaction techniques can be chosen whose dimensionality matches the requirements of the task.

*Select* tasks, although inherently one dimensional, generally require interaction techniques whose dimensionality matches that of the object being selected. So, in the case of the engineer selecting experimental components and engine types, if both the components and the engine types were represented as three-dimensional objects in the scene, the associated *Select* task would require a three-dimensional interaction technique. If, however, the components were represented as three-dimensional objects, and the engines were presented as items on a pull-down list, the *Select* task would require both 2D and 3D interaction techniques.

The dimensionality requirements of the *Position* and *Orient* tasks would also be closely linked to the objects being affected by the movement. Continuing with the same example, an engineer would need to position and orient a 3D representation of a new component in order to determine whether or not, or how well it would fit in an engine. Therefore the three-dimensional requirements of that task would drive the need for a three-dimensional interaction technique to accomplish it.

Given the description of this example application, it would be unlikely that the engineer would need to perform any *Path* tasks. However, it might be necessary to perform *Quantify* tasks if the engineer wanted to propose a new location or configuration for a new component so that it would fit in a given engine type. In this case, the dimensionality requirements of the task would be dependent on the type of *Quantify* task that the engineer needed to perform. Since this example requires the engineer to examine

new components for proper fit, a foreseeable task would require the engineer to make spatial measurements. Given the dimensionality of the environment and the objects in question, a 3D interaction technique would be best suited for accomplishing the task.

*Text* tasks, as described in the previous chapter, are inherently two-dimensional, thus requiring a two-dimensional interaction technique. The importance of the task to the overall goals of the application may have some impact on which technique is chosen to perform the task. However, as there are relatively few 2D interaction techniques currently available in VEs, *Text* tasks can often be the most challenging hurdle a VE application designer faces when trying to select appropriate interaction techniques and devices.

### 3. Task Prioritization

Clearly, the list of tasks that will result from the decomposition of any application down to its fundamental tasks will be quite long. Additionally, the number of techniques and associated devices would be greater than could be practically integrated into a single application. Therefore, it is necessary to prioritize the tasks with respect to the application that is being designed.

The primary intent of the engine design application is for the engineer to be able to view new components, place them in various engines, and write comments or recommendations. Therefore, the tasks of highest priority are those that enable accomplishing of that intent. They include selecting new components, selecting engines, positioning components, orienting components, and entering text comments or recommendations. These tasks are essential, since without them the application can not achieve its purpose. Other tasks may be included in the application to make it more

robust, and those also should be prioritized. However, as they would not be critical to the accomplishment of the intent of the application, they should not be classified as essential.

### 4. Technique and Device Selection

The VE application designer may find, after completing the task decomposition, dimensional categorization, and task prioritization, that although there are a range of 2D and 3D task requirements, the only dimensionality requirements of the essential tasks are either 2D or 3D. Should this be the case, these results will clearly point the designer to the environment, and thus the interaction techniques that are best suited for the application. If, however, there are essential tasks requiring both 2D and 3D interaction techniques, a few approaches to the design should be considered.

The application designer should first consider whether or not the dimensionality requirements of any of the essential tasks could be sacrificed for the overall functionality of the application. For example, if the application requires the user to perform a *Text* task, but that task is performed infrequently or the amount of text that must be entered is minimal, the designer might consider sacrificing the task's requirement for a 2D interaction technique. This may eliminate the need for multiple devices, thereby improving the overall functionality of the application and make it easier to use, since all interaction techniques would then be 3D and users would only require a single interaction device.

If, however, the essential tasks required both 2D and 3D interaction techniques, and sacrificing the dimensionality requirements of any of them would decrease, rather than improve the overall functionality of the application, then one of two options should

be considered. One approach would provide two sets of interaction devices; one set capable of accommodating the tasks requiring 2D interaction techniques, the other set capable of performing all necessary 3D interaction techniques. These sets could be a single device, such as a PDA or 3D mouse, or a collection of several devices, such as data gloves and a 6 DOF baton. This would enable the user to have access to the tools necessary to perform each task in a way that would match the dimensionality of the task to the technique.

In some instances, the presence of several interaction devices could prove to be too cumbersome, or actually hinder the overall usability of the VE application. In this case, the designer should consider using a hybrid device — one that is capable of performing both 2D and 3D interaction techniques. For instance, one might use a tracked PDA or Virtual Notepad, depending on whether the environment was presented using a CAVE or an HMD. The PDA or notepad could then be used for all 2D interaction, such as text entry or display, but could also be used to perform 3D interaction. A user could use a tracked PDA in a CAVE to point to objects, and then by a simple button push, select the object and change its position and/or orientation relative to changes made in the location and orientation of the PDA. The Notepad could be used in a manner similar to the Transparent Prop techniques or the Lifted Palm technique for changing the position and orientation of an object in the scene.

## C. SUMMARY

Regardless of the techniques and associated devices chosen, the most important issue for the application designer is to ensure that the devices, techniques, and tasks are matched in such a way that the overall performance and experience of the user is optimized.

# IV. METHODOLOGY

## A. INTRODUCTION

This chapter describes the methodology used to prove the thesis by providing an overview of the experiment, a discussion of the hardware and software used, and an explanation of the data collected. Results and analysis of the data will be discussed in Chapter V.

## B. EXPERIMENT OVERVIEW

The approach to virtual environment application design outlined in the previous chapter relies heavily on the hypothesis that matching the dimensionality of task requirements to interaction techniques improves task performance. In order to prove this hypothesis, it was necessary to conduct an experiment that examined performance on tasks of mixed dimensionality performed using both 2D and 3D interaction techniques. The task types chosen for the experiment were *Select*, *Position*, and *Text*. *Path* tasks combine *Position* and *Orient* tasks by introducing the element of time. The essential issues of task dimensionality requirements related to the performance of both *Orient* and *Path* tasks are covered sufficiently by the performance of *Position* tasks. Therefore, neither *Orient* nor *Path* tasks are evaluated in this experiment. *Quantify* tasks are also not examined, since they have no inherent dimensionality.

Upon beginning the experiment, the subjects read a brief overview of the experiment and signed consent forms (Appendices B-F). This was followed by a brief

demonstration of the VE application they would be using, thereby exposing them to the techniques used during the course of the experiment. Following the demonstration, subjects were presented with more material about the interfaces and the techniques that were used in order to reinforce procedures they had witnessed during the demonstration (Appendix G). The experiment began once they were satisfied that they understood the techniques.

Three interfaces were presented to each test subject. One interface contained only 3D interaction techniques; one contained only 2D interaction techniques; and the third was a hybrid interface possessing both 2D and 3D interaction techniques. To reduce the impact of a learning effect, the interfaces were presented to the subjects in different orders. There were six possible combinations of the three interfaces that were uniformly distributed among the test subjects. The first six test subjects received six different orderings of interfaces. The second six subjects received the same ordering as the first six, such that test subjects 1 and 7 experienced the three interfaces in the same order. Since there was a total of 27 test subjects, the first 24 experienced a uniform distribution of the six orderings as just detailed. The final three subjects were randomly assigned a task ordering without replacement.

The test subjects were read instructions from a script for each task they were to perform (Appendix H). They were allowed to ask questions if they did not understand any part of the instructions, but they were not allowed to begin execution of the task until all instructions had been read. An observer measured two values for each task: time

required and the number of errors committed. Following completion of all tasks with all interfaces, subjects were given a post-task questionnaire to complete (Appendix I).

Pilot tests showed that users often became confused about which technique had been used to perform each task with each interface. Therefore, subjects were provided with screen snapshots to remind them of what they saw and what techniques were used with each interface. This prevented the blurring effect that was discovered during pilot testing.

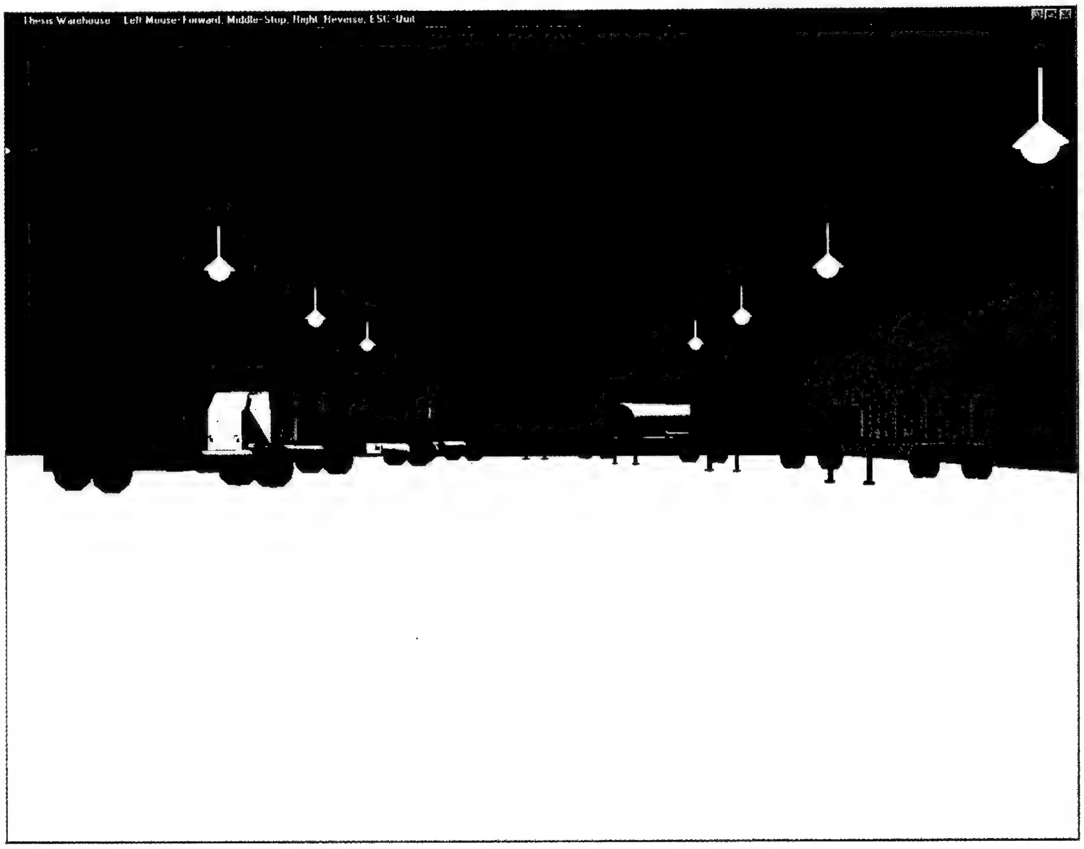



Figure 4.1. Virtual Warehouse Scene.

### 1. Select Tasks

The *Select* tasks required subjects to select objects in the scene based on spatial instructions. The scene presented to all test subjects consisted of a static view of the interior of a warehouse (Figure 4.1). On the left side, from the subject's viewpoint, was a row of four tractor trailer trucks. On the right side, there was a row of four trailers. Subjects were given spatial instructions directing them to select a specific truck. For instance, a subject might be instructed to select the third truck from the left. This was intended to eliminate any form of identification task. As the trucks were of different types and colors, subjects could have been instructed to pick the red truck or the Peterbilt 362E, however, this would have skewed the test so that it was no longer a test of a purely spatial task, but also an identification task. Providing the subjects with instructions that were spatial resulted in a test that could accurately determine if a dimensionality match between task requirements and interaction techniques was solely responsible for improved performance.

#### *a) 3D Interaction Technique*

As well as selecting a truck, subjects were also instructed to select a trailer that would later be positioned behind the truck. All subjects were required to use one of two interaction techniques to perform the *Select* task. The 3D interaction technique enabled subjects to use ray-casting to select objects. Subjects would point into the scene and select a truck or trailer, based on the spatial instructions they had been given. Once the object had been selected, the ray would disappear, and an acknowledgement would be displayed, providing them the name and color of the object that had been selected.

### b) *2D Interaction Technique*

The second interaction technique was a two-dimensional technique. Subjects were provided with a list, by name, of all the trucks and trailers in the scene. As with the 3D interaction technique, subjects were given verbal instructions such as, "Select the fourth trailer from the right." Based on those instructions alone, subjects were required to determine which item on the list was the fourth trailer from the right. Clearly, this was a more difficult technique for accomplishing a 3D task since a list does not provide any form of three-dimensional spatial information. In the implementation created for this experiment, all the trucks were different colors, and one could argue that by adding the color of each of the vehicles to the information in the list, the task would have been made easier. However, that would have combined an identification task with a selection task, thereby confounding the experiment. Furthermore, were all the vehicles the same color, the addition of such information to the vehicle names in the list would have provided subjects with no more assistance in performing the spatial task.

## 2. Position Tasks

The *Position* tasks required test subjects to move a trailer from one side of the warehouse to the other and position it directly behind a truck. Again, the instructions were spatial in nature. For example, subjects were instructed to move the third trailer from the right to a position directly behind the second truck from the left. Depending on the interface being used at the time, one of two possible interaction techniques was available for moving the objects. Regardless of the interaction technique that was used, subjects were required to position a trailer directly behind a truck, as instructed by the

observer. When the trailer was properly positioned and a subject indicated completion of the *Position* task, the trailer automatically hitched to the truck.

### *a) 3D Interaction Technique*

The 3D interaction technique closely resembled the HOMER technique discussed in Chapter II. Subjects would point into the scene using ray-casting, just as was done when performing the *Select* task using a 3D interaction technique. However, in this case, subjects would hold down a mouse button, much as is done when dragging and dropping an item in a desktop environment. As soon as the button was pressed, the ray disappeared and subjects gained control of the motion of the object, and its movement became hand-centered. The subjects' viewpoint was fixed, and no means was provided for navigation through the scene, however, the object, once controlled, moved in direct relationship to the location and heading of the subjects' hand. Object motion along the Y-axis was constrained to reflect realistic motion of a trailer across a warehouse floor. Additionally, orientation about the X-axis and Z-axis was also constrained for the purpose of task realism. One might argue that these constraints reduced the task to a two-dimensional task, however, as subjects had to physically change their position in the environment in order to change the location of the controlled object in the scene, the technique used had 3D properties. Furthermore, the control device held in the subjects' hand, and therefore the subjects' hand motions, were not constrained to a single plane. A further argument could also be made that the introduction of movement in a direction perpendicular to the display surface gives the perception and sense that the movement is 3D, thus requiring a 3D technique.

### *b) 2D Interaction Technique*

The 2D interaction technique presented subjects with a 2D display on a hand-held tablet containing two slide bars with egocentric directions (Figure 4.2). Subjects used a stylus to manipulate the slide bars, which in turn moved the selected object in the scene. The slide bars coincided with movement of the object in the XZ plane, thereby constraining movement along the Y-axis as in the 3D technique. The 2D

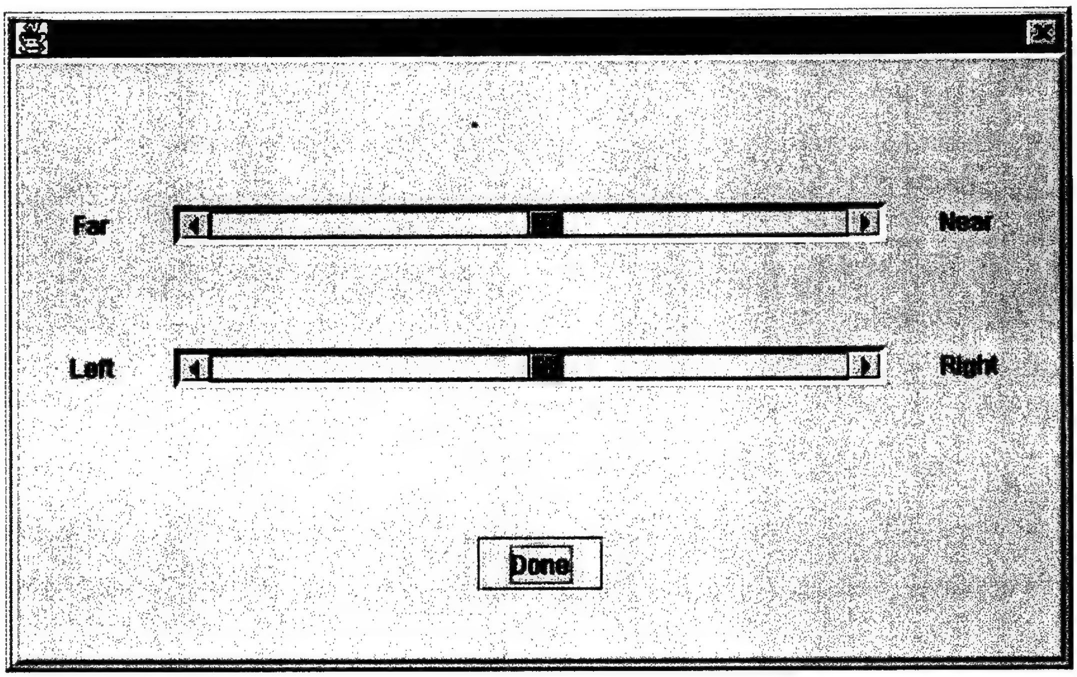



Figure 4.2. Interface for 2D *Position* Interaction Technique.

technique also did not provide any means for changing the orientation of the object in heading, pitch, or roll, in essence simplifying the 2D task over the 3D task, since the 3D technique enabled the subject to change the object's heading. The trucks and trailers were positioned in the scene such that their heading was identical, thereby eliminating the need for the subject to make any adjustments in the heading of the object being moved.

Despite these constraints, the task still had 3D requirements for the same reasons as discussed previously for the 3D technique.

### 3. Text Tasks

The *Text* tasks required subjects to perform a simple text entry. Subjects were instructed to enter the year of their birth so that it could be displayed as a vehicle identification number on the side of one of the trucks in the scene. A second task required subjects to display textual data about the truck or trailer that was selected, read the data, and provide the observer with some specific data from what they read. Since *Text* tasks are inherently 2D, these two tasks tested both the techniques used to input 2D symbolic data and the techniques used to display the same type data. As with the *Select* and *Position* tasks, subjects were required to perform the *Text* tasks using one of two interaction techniques.

#### *a) 3D Interaction Technique*

The 3D techniques associated with the entry and display of 2D text are representative of some common techniques currently used in VE applications. The technique used for displaying textual information about the vehicles in the warehouse required subjects to use the stylus to tap a button on an interface on the hand-held tablet. The data was then displayed as floating text in the environment (Figure 4.3). Although this technique allowed subjects to continue to view the elements of the environment behind the floating text, the text tended to blend in with the background and became difficult to read. Displaying the text on a floating window would have eliminated the

problem of the text blending in with the background, but it would have also obscured the objects in the scene about which the data was being displayed.

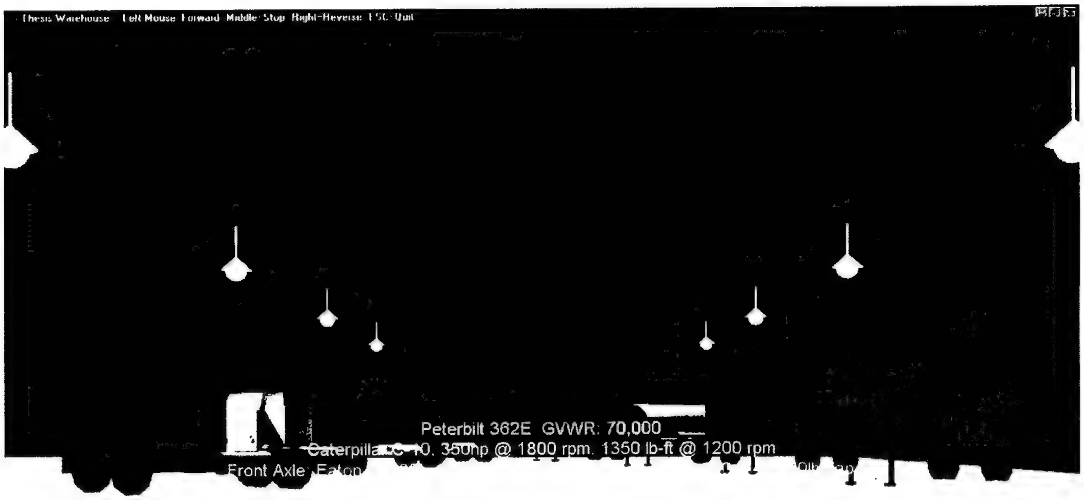



Figure 4.3. 3D Technique for Displaying Data.

The 3D technique for entering text presented subjects with a series of number squares that they could point at using the same technique used to select vehicles in the scene (Figure 4.4). Subjects were required to use the virtual number buttons and the ray-casting selection technique to enter the year they were born. Each number appeared as a vehicle identification number on the side of the green truck. Had a subject made an error entering the year, a backspace button was provided so that corrections

could be made. Once subjects finished entering the year, the "Done" button was used to remove all the number buttons from the scene. An alternate implementation would have presented subjects with an object-center or a floating window requiring a 3D pointing technique to select each digit of the year from a pull-down list containing the numbers 0 - 9. A technique similar to this was used in the CHIMP implementation of control panels

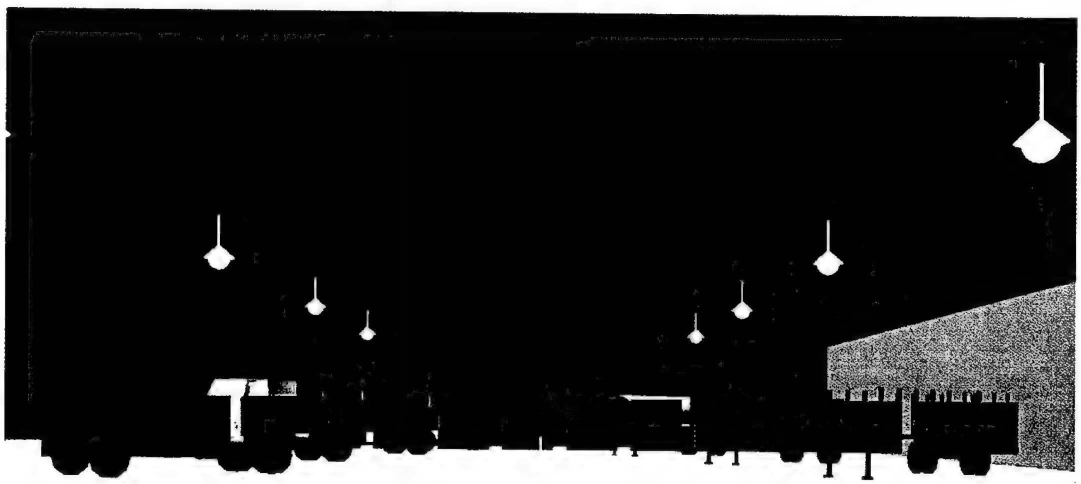


Figure 4.4. 3D Technique for Entering a Number.

(Mine, 1996). Both techniques required subjects to perform a 2D task using a 3D interaction technique.

### *b) 2D Interaction Technique*

The 2D interaction technique requires subjects to use the stylus to press a button on the interface on the hand-held computer. The textual data was displayed to the experimental subjects in a text box in a Graphical User Interface (GUI) displayed on the hand-held computer. The same data was displayed in the text area that would be presented in the environment using the 3D technique (Figure 4.5).

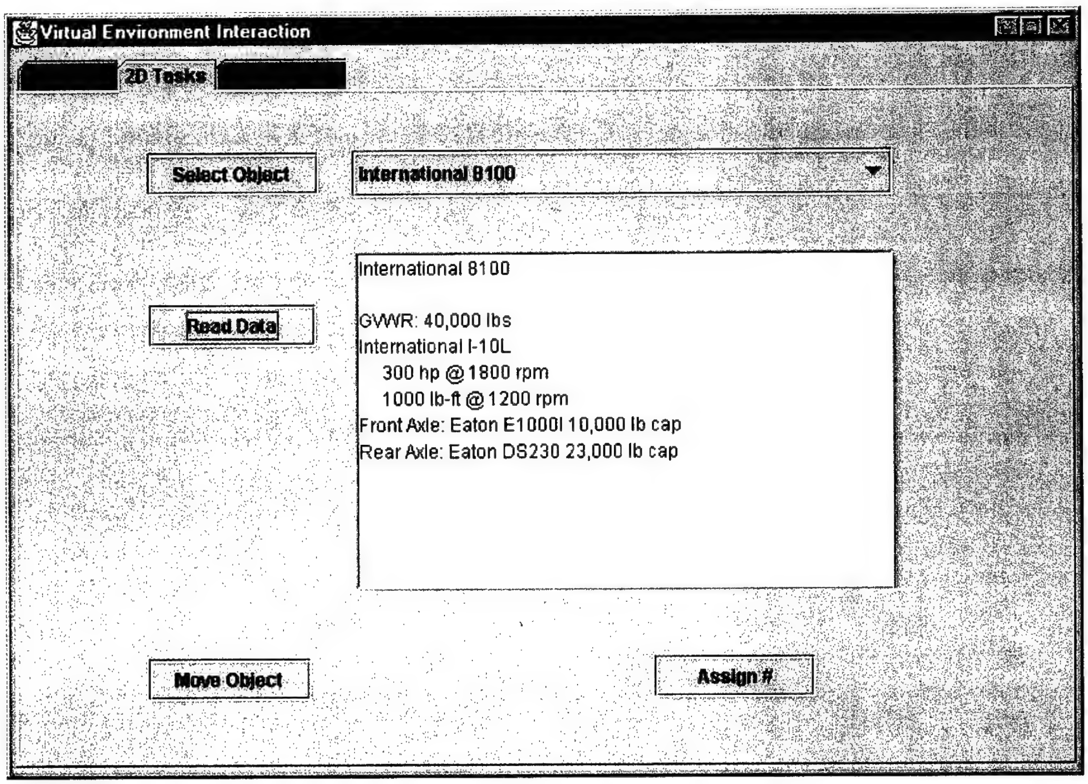



Figure 4.5. 2D Display of Textual Data.

The 2D technique for entering text makes use of the Microsoft PenWindows capabilities resident on the hand-held computer. Subjects used the stylus to tap a button on the interface, thereby displaying a small dialog box containing a text field.

Tapping a prompt in the text field displayed a screen keyboard, enabling the subject to use the numbers on the keyboard to enter the year they were born. After entering the number and pressing the "Done" button on the dialog box, the dialog box disappeared and the number was displayed on the side of one of the trucks in the scene.

## C. IMPLEMENTATION

A variety of hardware and software packages were required to create the environment, the interfaces, and the interaction techniques necessary to run this experiment. Hardware selection prioritized availability, then cost. The software used to design the virtual environment, Vega™, was selected because it is commonly used for VE design, it provides a wide range of device libraries, it enables real-time interaction, and it was readily available. The interfaces were programmed in Java because of the language's inherent networking capabilities and the ease with which GUIs can be designed.

### 1. Hardware Components

The design of this experiment, especially its requirements for both 2D and 3D interaction techniques, included the use of several pieces of hardware. Following is a description of all the components required for the interfaces and interaction techniques associated with this experiment.

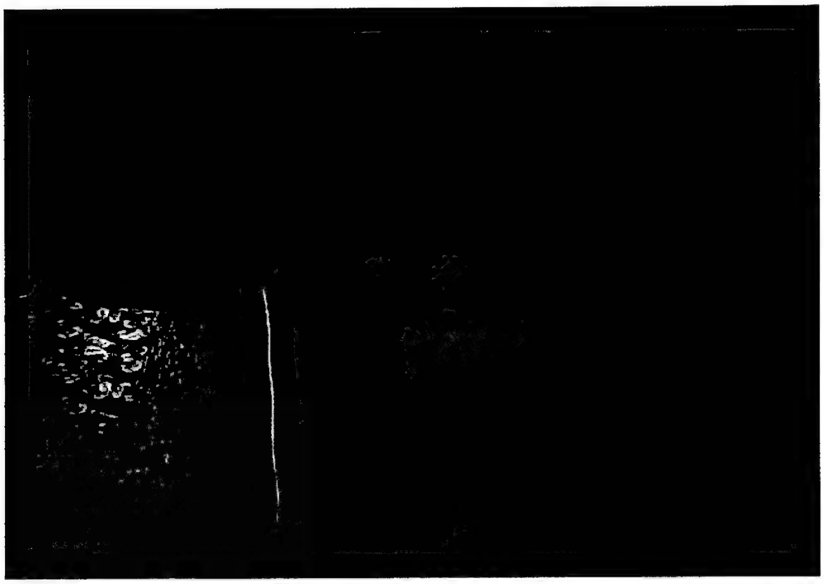


Figure 4.6. Author in the MAAVE.

***a) MAAVE***

The Multi-Angled Automatic Virtual Environment (MAAVE), created by Christianson and Kimsey (2000), served as the display system for this experiment (Figure 4.6). The MAAVE is a large, three-screen Virtual Environment Enclosure (VEE).

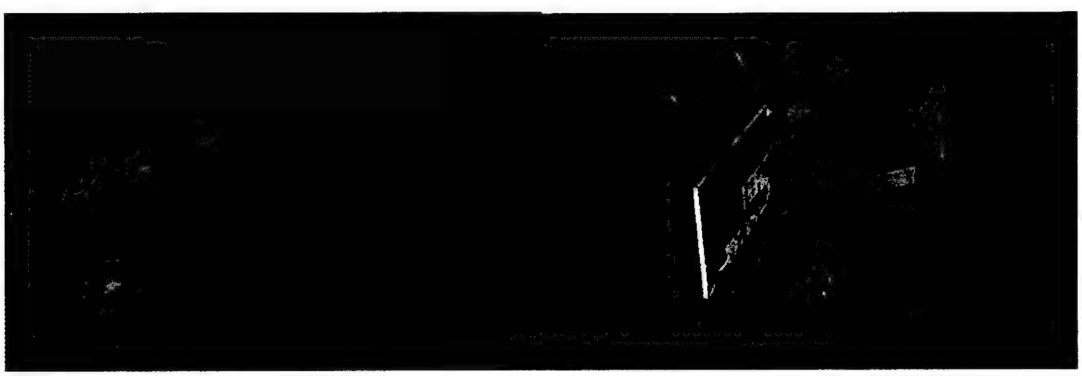


Figure 4.7. MAAVE Configuration.

The three rear projection screens are 5 feet by 7 feet each and are placed at a 135 degree angle from one another (Figure 4.7). The VE is displayed on the screens using three

stereo-capable VRex 2210 projectors. The computer driving the MAAVE is an Intergraph TDZ2000 GL2 running Windows NT 4.0. It has dual Pentium 400 processors and 512MB RAM. Three Wildcat 16MB video cards are used to produce the combined 3840 x 800 resolution display.

### *b) Hand-held Computer*

The hand-held computer used in the VEE was a Fujitsu Stylistic 1200 tablet (Figure 4.8). It served as the 2D interface between the test subject and the virtual

Figure 4.8. Fujitsu Stylistic 1200 Hand-held Tablet.

environment. It runs Microsoft Windows 95 and has a Cyrix 180 MHz processor, 64MB of EDO RAM, and a 640 x 480 VGA display screen. The tablet uses a WaveLAN

Bronze care for wireless network communications with the Intergraph machine driving the MAAVE applications. The primary interaction device used with the tablet was a proximity sensitive stylus.

### c) *Polhemus Fastrak*

A Polhemus Fastrak magnetic tracking device was incorporated into the VE design in order to provide spatial position and orientation data to the VE application. A Polhemus Long Ranger™ was used in conjunction with the Fastrak to provide a 30 foot range with 4 ms latency at 120 Hz. An advantage to using the Polhemus Fastrak was that it could be easily incorporated into most VE applications and was a standard device type that is supported by Vega.

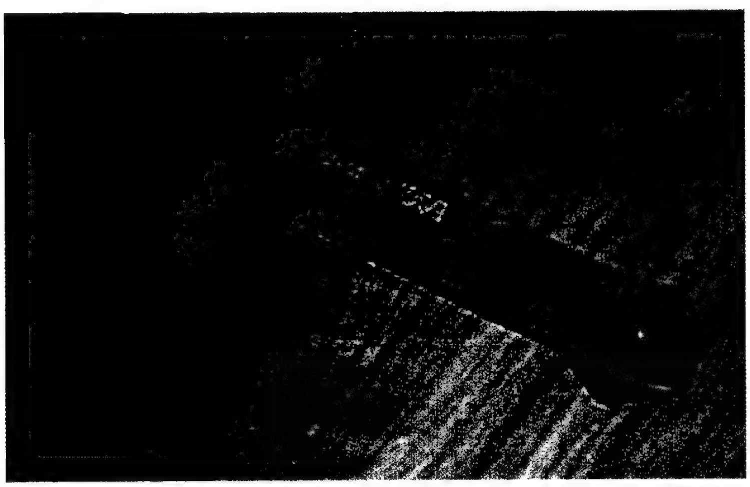


Figure 4.9. Mouse Pen.

### d) *Mouse Pen*

The Mouse Pen is a two-button serial mouse created by Questec™ and is shaped much like a pen (Figure 4.9). For use in this experiment, it was attached by its six foot serial cable to the tablet and was used as a pointing device for 3D interaction

techniques. A hybrid device was created by taping the stylus and the Mouse Pen together so that they faced opposite directions. The Polhemus Fastrak receiver was also taped to the stylus / Mouse Pen combination so that the motion of objects in the scene that were selected using the Mouse Pen would become hand-centered. The need for a pen-shaped, serial mouse device arose from the Java implementation used for the GUI. In order for the Java application on the tablet to register a mouse click when the subject was pointing into the scene, it was necessary to use a device that was physically connected to the tablet. The Mouse Pen provides that physical connection, while also providing a device whose shape is similar to the stylus, and is therefore a comfortable and practical device to use for 3D interaction with the environment.

#### *e) Apple AirPort*

The Apple AirPort™ enabled wireless network communication between the interfaces running on the tablet and the VE application running on the Intergraph. The AirPort is a DHCP server operating at 11Mbps. The tablet acquired its IP address from the AirPort while booting, enabling it to perform TCP communications with a server embedded in the VE application. If no static IP address is required for communications with other devices, the AirPort can be used immediately upon removing it from its packaging, with no software configuration requirements. If static IP addresses are required, or any other unique networking situations exist, the AirPort must be configured using an Apple computer.

### 2. Virtual Environment

The 3D virtual environment needed for this experiment was created using Vega and Creator. Multigen Creator was used to create all the models used in the environment. These included the warehouse, the environment outside the warehouse, and the truck and trailer models inside the warehouse. No attempt was made to make the models look photo-realistic. This experiment was concerned with subject performance on spatial tasks, rather than the degree to which the subject felt immersed in the environment.

LynX is a GUI tool for designing Vega applications; it was used to create the initial virtual environment for this experiment. LynX provided an easy method for importing models, configuring lighting, creating motion-models for navigation and object movement, establishing players, configuring input devices, building text objects for textual data display, and constructing isectors to be used for collision detection between trucks and trailers. LynX configuration settings are stored in a data file and can be used to configure and launch a Vega application using a bootstrap program written in C (provided with Vega).

The provided bootstrap program required modification to support the hand-held tablet interface. A server was embedded in the Vega application, and a simple protocol was developed to enable quick TCP communications that required a minimal amount of data transfer. The application was constructed to be stateful, thereby enabling number codes to be used for passing information from the tablet to the VE application about state changes, and to indicate when integer or float data was being transferred (Table 4.1). To prevent significant degradation in the frame rate, the VE application checked for new

| PROTOCOL | DEFINITION |
|---|---|
| 0 | Clear all values |
| 1 | State Change |
| 1 0 | State Change: Reset |
| 1 1 | State Change: 3D Select |
| 1 2 | State Change: 3D Move |
| 1 3 | State Change: 2D Select |
| 1 4 | State Change: 3D Read |
| 1 5 | State Change: Hybrid Select |
| 1 6 | State Change: 2D Move |
| 1 7 | State Change: 3D Assign # |
| 1 8 | State Change: 2D Assign # |
| 1 9 | State Change: Auto Hitch |
| 2 | Mouse Event: |
| 2 0 | Mouse Event: Mouse Down |
| 2 1 | Mouse Event: Mouse Up |
| 3 | Integer Data |
| 3 i | Integer Data: Integer Value |
| 4 | Float Data: |
| 4 fx | Float Data: Float X-value |
| 4 fy | Float Data: Float Y-value |

Table 4.1. Application Protocol.

messages only every fourth frame. Message checking frequency was altered during run-time to allow for more frequent data transfer during latency-sensitive tasks like *Position.* Additionally, all objects, players, isectors, input devices, and motion-models defined in the data file needed to be instantiated as objects in the C code so that data could be extracted from and assigned to them regarding their position, orientation, and associations with other objects.

### 3. Interface Design

The interface between experiment subjects and the virtual environment was a Java application running on the hand-held tablet. A client was created to enable communications with the server embedded in the VE application. Each element of the GUI that could affect elements of the VE scene had an embedded client object, enabling

changes indicated by interaction with the GUI to be communicated with the Vega application.

As seen in Figure 4.10, there were three interface tabs on the tablet GUI. One tab presented subjects with only 3D interaction techniques; another tab provided only 2D interaction techniques; and a third hybrid tab enabled use of a combination of 2D and 3D interaction techniques. The 3D tab consisted of five buttons: a "Select Object" button, a "Read Data" button, a "Move Object" button, an "Assign #" button, and a "Quit" button. (Figure 4.10). Each of the first four buttons, activated a 3D interaction technique. The "Select Object" button made a ray appear in the scene, enabling the test subject to use

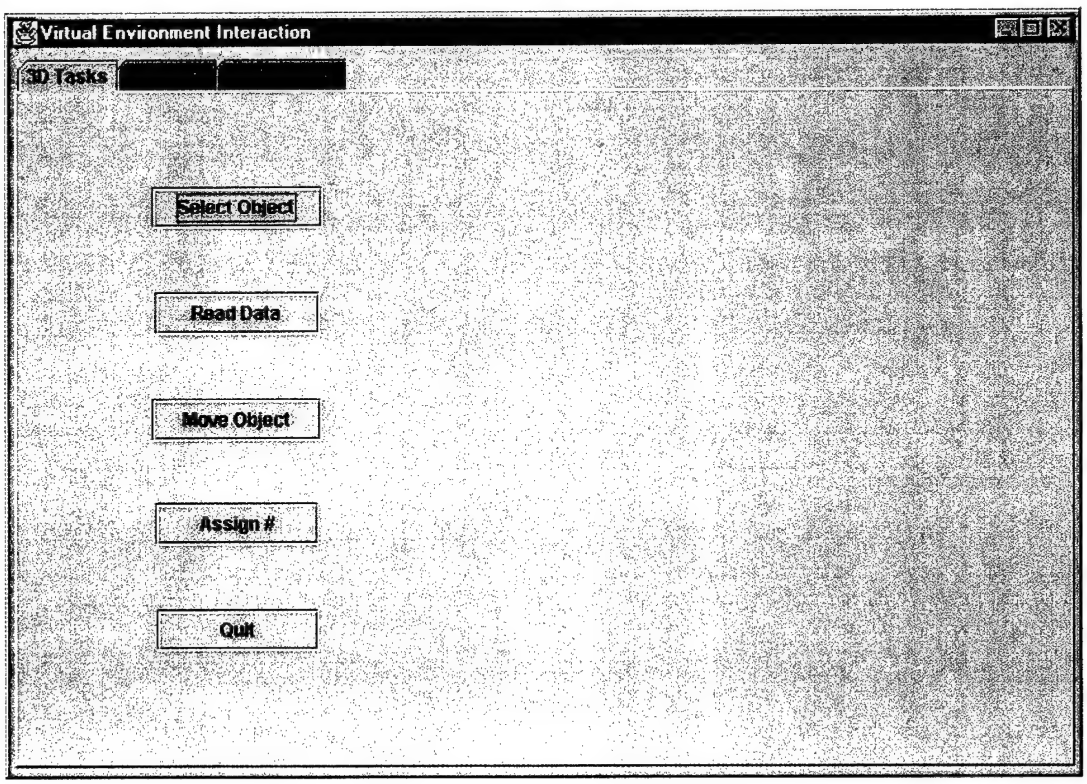



Figure 4.10. 3D Interface.

ray-casting to select objects in the scene. The "Read Object" presents textual data as floating text in the scene. The "Move Object" button activates the ray in the scene that remains until an object had been selected and its motion has become hand-centered. The "Assign #" button activates the number buttons and the ray so that they are all displayed in the scene, enabling subjects to use ray-casting to enter a number.

The 2D interface presented the subject with a button and a pull-down list for selecting an object, a button and a text area for reading data, a button and dialog box containing slide bars for moving an object, and a dialog box with an associated screen keyboard for entering a number to be used as a vehicle identification number (Figure

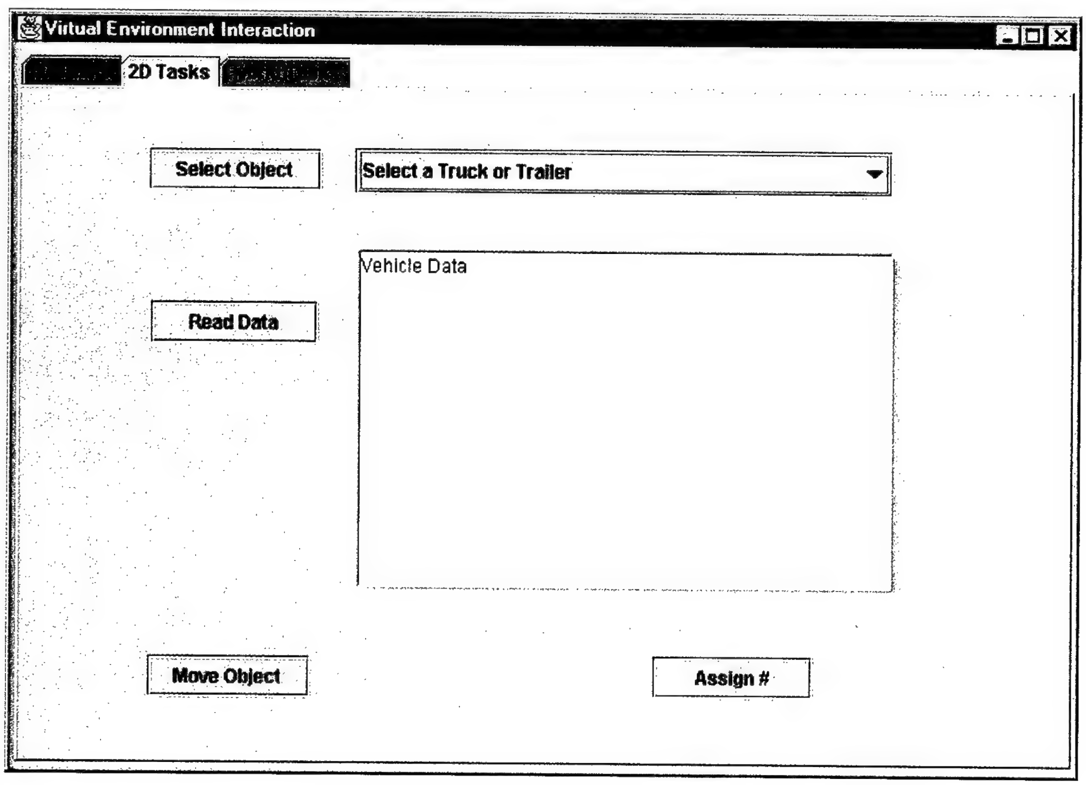



Figure 4.11. 2D Interface.

4.11). To perform the select task, subjects were provided with spatial instructions and were required to use the pull-down list to select the directed object or objects from the scene. Tapping on the "Read Data" button displayed data about the selected object in the text area. Pushing the "Move Object" button displayed a dialog box to the subjects, forcing them to attempt to move the selected object through the scene so that it could be connected to a truck, as directed. The "Assign #" button revealed a dialog box with a text field capable of accessing the screen keyboard. The keyboard was used to enter the number directed by the observer, and the "Done" button was pressed when all interaction was completed.

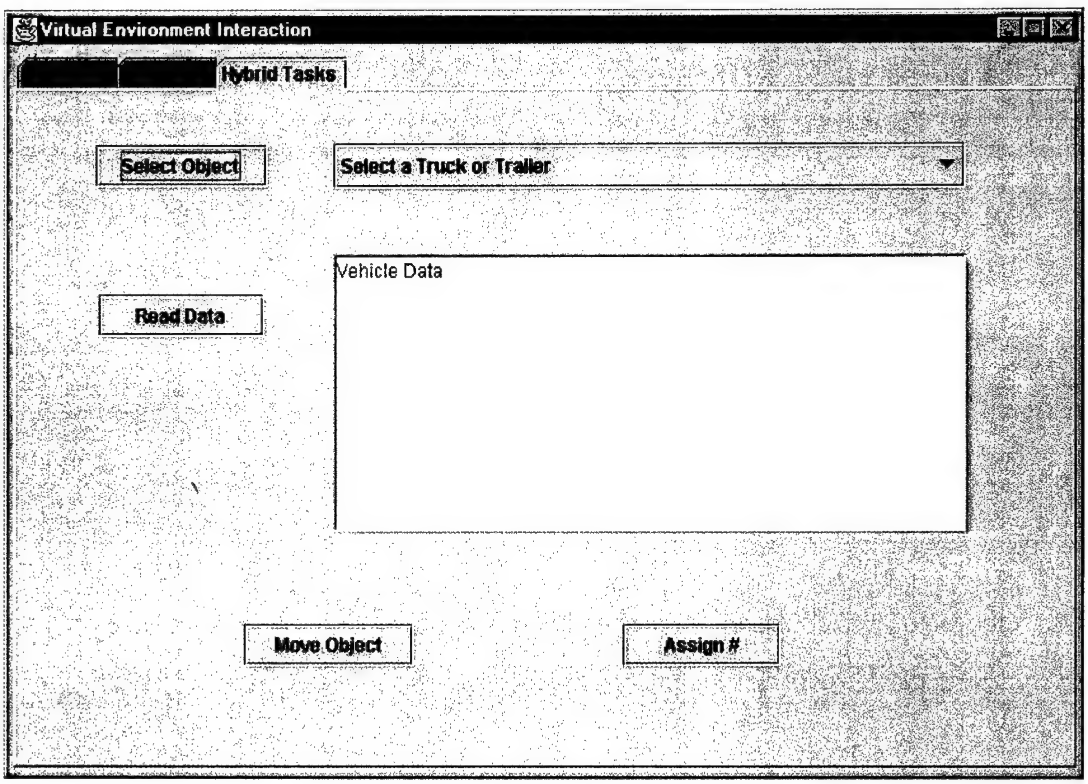



Figure 4.12. Hybrid Interface.

The Hybrid interface presented subjects with both 2D and 3D interaction techniques (Figure 4.12). Like the other two interface tabs, this one provided experiment subjects with techniques to perform the *Select*, *Position*, and *Text* tasks performed in the other interfaces. The *Select* task could be preformed using either a 2D or a 3D interaction technique. Subjects could either press the "Select Object" button and point into the scene using the Mouse Pen, or they could select an object using the pull-down list. The "Read Data" button and the "Assign #" button provided subjects with the same techniques that were used in the 2D interface. Additionally, the "Move Object" button on the hybrid interface also enabled subjects to perform the task of moving a trailer behind a truck using a 3D interaction technique, identical to the one performed when using the 3D interface.

## D. PERFORMANCE MEASURES

Quantifiable measurements were necessary to assess performance when dimensionality matches and mismatches occurred. To that end, each subject's performance was evaluated in order to record the amount of time required to accomplish each task and the number of errors committed in the course of completing each task. Additionally, data was collected using a post-task questionnaire to elicit feedback from the experiment subjects regarding their opinion of the ease or difficulty of accomplishing each task using the interaction techniques presented. Subjects were also asked for their overall interface preference: 3D, 2D, or hybrid.

### 1. Time

The first quantifiable performance measurement was the time required for task completion. In order to maintain the integrity of the results, uniform start and stop time indicators were established for each task. Furthermore, the observer read the instructions aloud for each task during the experiment, thereby reducing possible variability resulting from different reading abilities and speeds. After being read the instructions for a given task, subjects were allowed to ask questions to clarify the task. Task start and stop time indicators differed based on the interaction technique employed, and thus, the interface that was used. Each task, technique, and associated start and stop time indicators will be discussed in detail.

#### *a) Select Tasks*

The *Select* task employed different interaction techniques for each interface that was presented. When subjects used the 3D interface, they were required to use the stylus to press the "Select Object" button on the hand-held tablet. This activated the ray in the scene, enabling them to point and select an object from the scene given spatial instructions. Time started for this task when the subject tapped on the "Select Object" button. Time stopped when the subject selected the correct object from the scene and a dialog box appeared on the tablet screen providing the name and color of the truck or trailer that had been selected.

When subjects used the 2D interface, the *Select* task required them to use the stylus to tap on a pull-down list on the tablet display, pick a vehicle from the list, and then tap the "Select Object" button. Time for this task started when the subject tapped on

the pull-down list and stopped when a dialog box appeared on the tablet display, indicating the correct object had been selected from the scene.

The hybrid interface provided the subjects with both the 2D and 3D interaction techniques. As they were required to select both a truck and a trailer at different points while using each interface, subjects were instructed to select a given truck using the 3D interaction technique, and the trailer using the 2D interaction technique. This exposed them to a contrast in techniques within the same interface and helped eliminate any blurring that may have occurred between the two techniques. The start and stop times for each *Select* task were the same as outlined above, depending on the dimensionality of the technique being used.

#### *b) Position Tasks*

The *Position* task employed the same interaction technique in both the 3D and the hybrid interface. Using these interfaces, subjects were required to use the stylus to press the "Move Object" button, activating the ray and enabling them to point into the scene and move a truck or trailer using a 3D technique. The time start indicator for the 3D interaction technique was the "Move Object" button push. Time stopped when the correct trailer was hitched to the correct truck. If no connection occurred, subjects had to press the "Move Object" button again and continue to adjust the trailer position until the two objects hitch together properly.

The 2D interface presented subjects with a 2D interaction technique for moving a selected trailer. Subject were required to press the "Move Object" button, activating a dialog box with slide bars that enabled movement of the trailer using

egocentric directions. Time for this task began when a subject used the stylus to tap the "Move Object" button and stopped when the trailer was hitched to the correct truck. Just as in the 3D technique, subjects had to continue to make adjustments to the location of the trailer until it connected with the truck.

#### *c) Text Tasks*

The *Text* tasks employed the same interaction technique in both the 2D and the hybrid interface. Using these interfaces, subjects were required to perform two distinct tasks. They read textual data that was displayed about the selected object and were also instructed to enter the year they were born so that it could be displayed as a vehicle identification number on the side of one of the trucks in the scene. In order to read data about the selected object using the 2D interface provided with the 2D and hybrid interfaces, subjects tapped on the "Read Data" button, causing the textual information to be presented on the tablet display. The instructions for this task also directed them to find a specified fact about the selected object from the textual data and convey that information to the observer. The time start point for this task occurred when the subject used the stylus to press the "Read Data" button on the interface. Time stopped when the subject correctly provided the observer with the requested information about the selected object.

The task of entering a number using the 2D and hybrid interfaces required the subject to tap the stylus on the "Assign #" button. A dialog box appeared enabling subjects to access a screen keyboard with which the subject's birth year was entered

(Figure 4.13). Time started for this task when subject hit the "Assign #" button on the

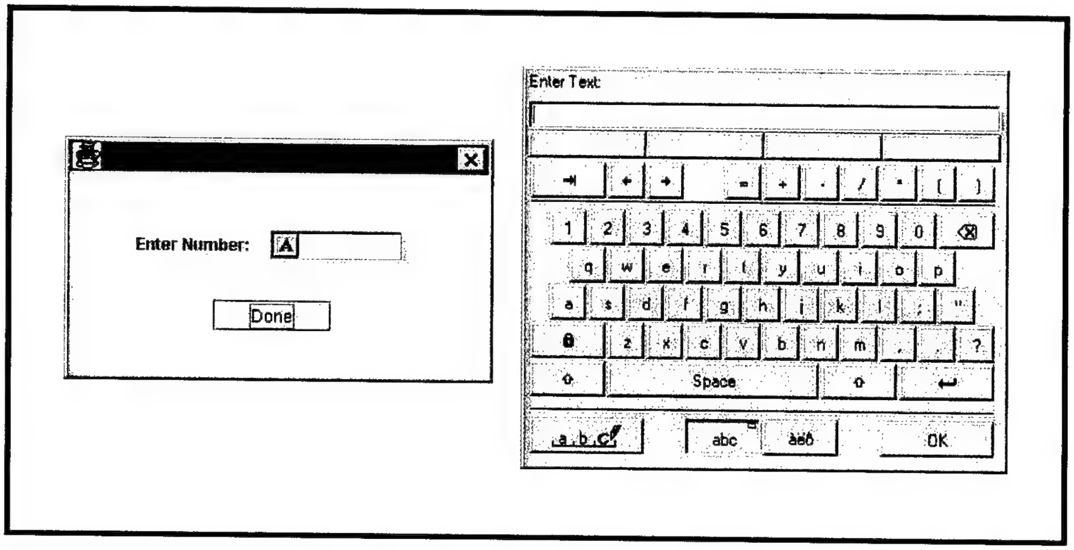



Figure 4.13. Assign # Dialog Box and Associated Screen Keyboard.

interface. Time stopped when the correct number appeared on the side of one of the trucks in the scene. This task did not require the subject to select which truck would be assigned the vehicle identification number.

When subjects were presented with the 3D interface, 3D interaction techniques were used to perform both *Text* tasks. In order to read data about the selected vehicle, subjects pressed the "Read Data" button on the 3D interface. The textual data was then displayed in the scene. Just as with the 2D and hybrid interfaces, subjects were required to provide the observer with a specific data item extracted from the displayed text. The start time for this task occurred when a subject clicked on the "Read Data" button and ended when the subject provided the observer the requested data.

The text entry task on the 3D interface was activated by the subject tapping on the "Assign #" button. Number buttons and a ray appeared in the scene, and

subjects used a 3D selection technique to enter numerical data. The task started when a subject pressed the "Assign #" button and concluded when the "Done" button was selected and all the number buttons disappeared from the scene.

### 2. Accuracy

The second quantifiable performance measure was accuracy, determined by the number of errors committed by the test subject in performing each task. Again, to maintain the integrity of the results, standard definitions for errors were established for each task. Error definitions for each task were identical regardless of the interface or interaction technique used to perform the task. The following are simple descriptions of possible errors for each task.

#### *a) Select Tasks*

*Select* task errors occurred when subjects selected the wrong object from the scene. When the 3D interaction technique was employed, errors include missing the object when using ray-casting to select it or selecting the wrong truck or trailer based on the spatial instructions provided. When using the 2D interaction techniques, selecting the wrong truck or trailer from the pull-down list was counted as an error.

#### *b) Position Tasks*

*Position* task errors were noted when the subject failed to position the trailer behind the correct truck or when the truck and trailer failed to automatically connect. These errors could occur regardless of which interaction technique was being employed.

#### c) *Text Tasks*

*Text* task errors were measured for both the reading task and the text input task. A reading error was annotated when subjects provided incorrect data to the observer. A text input error occurred when an incorrect digit was entered and had to be corrected. When using the 3D interaction technique, *Text* errors occurred when subjects missed a number button with the ray or selected an incorrect number. When using the 2D interaction technique, an error was annotated when the test subject tapped an incorrect number on the screen keyboard.

### 3. User Preference

A subjective measurement used in this study was user feedback in the post-experiment questionnaire (Appendix I) . Each subject was presented with the questionnaire following completion of all tasks in the experiment, and was asked to rate the degree of ease or difficulty afforded them in performing each task using the interaction techniques provided with each interface. These questions were not intended to elicit opinions about the interface design. Instead they were meant to obtain feedback about the interaction techniques used to perform each of the tasks in each of the interfaces. Tasks were rated on a scale of 1 – 5, using the following values for each score:

1. Difficult
2. Somewhat Difficult
3. Neutral
4. Somewhat Easy
5. Easy

The questionnaire also elicited opinions about which overall interface and which interaction techniques were preferred for each of the four tasks. Subjects were asked to explain their answers briefly.

THIS PAGE INTENTIONALLY LEFT BLANK

# V. RESULTS AND ANALYSIS

In order to prove that matching the dimensionality of task requirements to interaction techniques will improve performance on tasks as opposed to when a mismatch occurs, the experiment outlined in Chapter IV was conducted with 27 unpaid test subjects participating. The subjects were students and staff members at the Naval Postgraduate School ranging from age 19 to 41. Computer experience ranged from 1 to 20 years, and 20 of the 27 subjects had some prior experience in a VE. All test subjects rated themselves as having at least a moderate level of computer expertise, and 15 of the 27 subjects evaluated themselves as possessing average to above average skill with computer games.

## A. PERFORMANCE RESULTS

Data was collected on the performance of all 27 subjects for the *Select*, *Position*, and *Text* tasks. Given the hypothesis of this thesis and support demonstrated by Lindeman's work (1999), the expectation is that the data collected from this experiment will show a significantly better task performance when the dimensionality of the interaction technique used by the subjects matches the dimensionality requirements of the task they are performing. Following are the results and data analysis from the experiment.

### 1. Select Task

The *Select* task required subjects to select an object from the scene using either a 2D or a 3D technique. The *Select* task was performed twice during each interface

presentation; however, the second time the *Select* task was performed, it was combined with instructions to read data about the selected object. The Select/Read combination task will be treated as a separate subset of the *Select* task data, as Read times and errors using a 2D interface (analyzed separately under *Text* tasks) were small enough to indicate that the *Select* task portion was the driving factor behind the performance results on this task. Subjects used a 3D technique while working with the 3D Task interface, they used a 2D technique while working with the 2D Task interface, and used both techniques while utilizing the Hybrid Task interface.

| | Count | Mean | Std. Dev. | Std. Err. |
|---|---|---|---|---|
| 2D | 27 | 56.926 | 28.095 | 5.407 |
| 3D | 27 | 21.519 | 21.063 | 4.054 |
| Hybrid | 27 | 16.111 | 6.296 | 1.212 |

| | Mean Diff. | Crit. Diff | P-Value | |
|---|---|---|---|---|
| 2D, 3D | 35.407 | 11.160 | <.0001 | S |
| 2D, Hybrid | 40.815 | 11.160 | <.0001 | S |
| 3D, Hybrid | 5.407 | 11.160 | .3377 | |

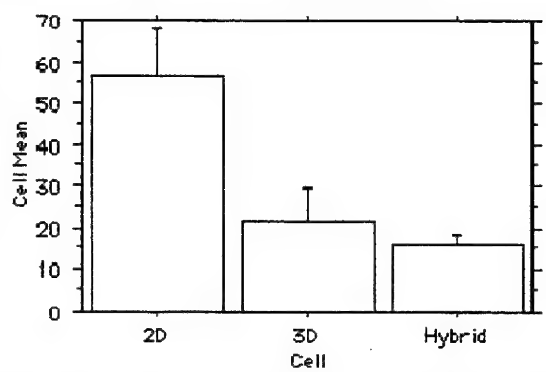



Figure 5.1. Time Results for *Select* Task.

### *a) Time*

The first performance measurement for the *Select* task is the time required to perform the task. Subjects used a 3D interaction technique with both the 3D and Hybrid interfaces, and a 2D interaction technique with the 2D interface. By examining the data in Figure 5.1, it is clear that the results were as expected. The average time required to perform the *Select* task using a 2D interaction technique was more than twice

the average time required using the 3D interaction technique. Furthermore, the p-value for the 2D/3D and 2D/Hybrid comparisons indicates that there is statistical significance between the 2D interaction technique results and the 3D interaction technique results. These results do not indicate that 3D interaction techniques are better than 2D interaction techniques. They only show that 3D interaction techniques are better for tasks whose dimensional requirements are 3D. Had the task been structured such that subjects were instructed to select the Peterbilt 362E or the Trailstar ADFT, then a 2D technique would have been best, since in that case, the 3D scene does not provide the 2D textual data required to perform the task. A 2D technique would provide that necessary data.

| | Count | Mean | Std. Dev. | Std. Err. |
|---|---|---|---|---|
| 2D | 27 | 38.556 | 15.106 | 2.907 |
| 3D | 27 | 26.074 | 7.395 | 1.423 |
| Hybrid | 27 | 60.111 | 36.152 | 6.957 |

| | Mean Diff. | Crit. Diff | P-Value | |
|---|---|---|---|---|
| 2D, 3D | 12.481 | 12.473 | .0499 | S |
| 2D, Hybrid | -21.556 | 12.473 | .0009 | S |
| 3D, Hybrid | -34.037 | 12.473 | <.0001 | S |

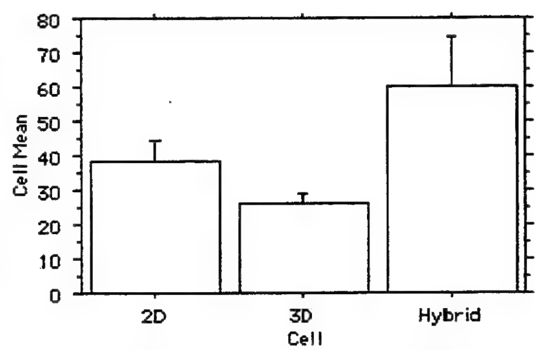



Figure 5.2. Time Results for *Select/Text* Task Combination.

The data in Figure 5.2 reflects the average time requirements to complete a more complex task that combined a *Select* task with a *Text* related task. Subjects were required to select an object from the scene given spatial instructions, display and read data about the object, and convey a specific piece of information to the observer from what they read. The 3D interface was the only interface that presented subjects with a 3D

interaction technique for selecting objects in the scene. Therefore, as expected, subjects needed less time to select objects in the scene when using an interface that provided a 3D interaction technique than with the other interfaces.

Both the 2D and Hybrid interfaces presented the subjects with a 2D interaction technique. Based on this fact and uniform distribution of the order in which interfaces were presented to the subjects, one would expect these two values to be very similar. However, the data shows a statistically significant difference between the average times with each interface. The reason that such a difference occurs between the 2D and Hybrid interfaces is due to a learning effect.

There were a total of eight objects in the scene, and therefore the list of objects presented with the 2D interaction technique was short. It is reasonable to expect that fewer errors would occur in later selections due to the narrowed possibilities. The Select/Read task was the third task performed with each interface. Therefore, when subjects saw the 2D interface before the Hybrid interface, they were able to use the 2D interaction technique once before having to perform the Select/Read task, since the first *Select* task performed with the 2D interface utilized the 2D interaction technique. However, when subjects used the Hybrid interface before using the 2D interface, the Select/Read task was the first time they experienced the 2D interaction technique for selecting an object, as the first *Select* task with the Hybrid interface utilized a 3D interaction technique. This means that when subjects performed the Select/Read task associated with the 2D interface, it was either the second or third time they had used the

2D interaction technique to perform the *Select* task. On the other hand, when half of the subjects performed the Select/Read task with the Hybrid interface, it was the first time

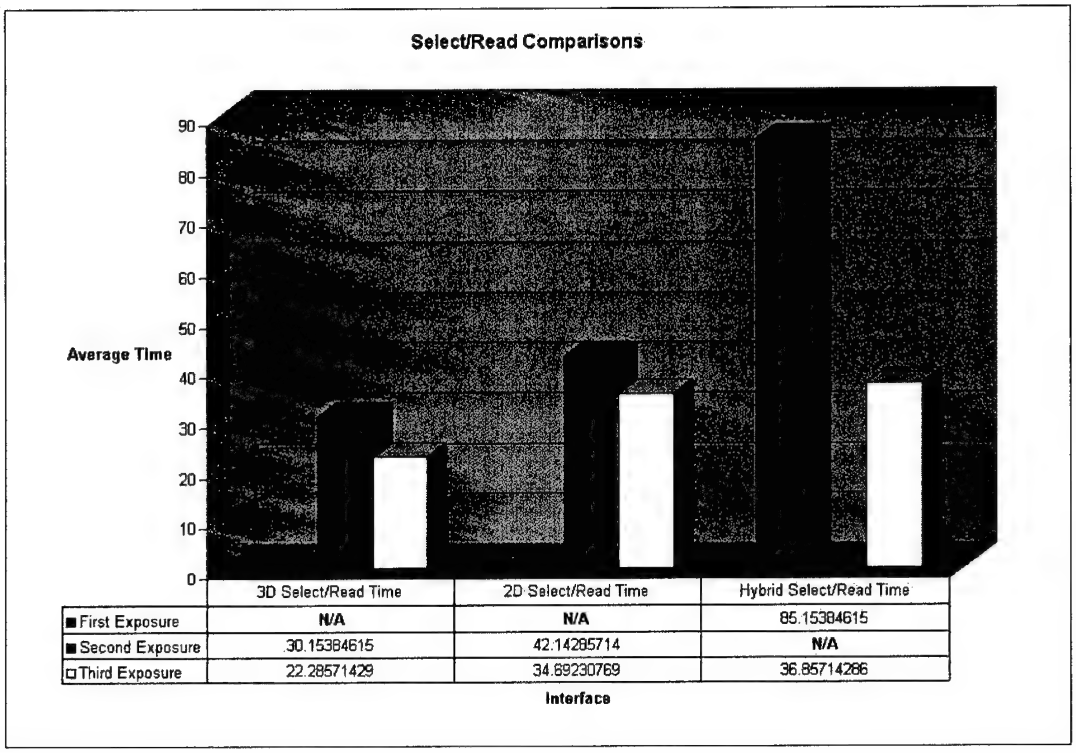



Figure 5.3. 2D/Hybrid Interface Comparison of Select/Read Task.

they were using the 2D interaction technique to perform the *Select* task. Figure 5.3 reflects these facts and clearly shows why, on average, subjects needed more time to perform the Select/Read task with the Hybrid interface. It is interesting to note that the average time required was almost the same with both the 2D and Hybrid interfaces when the Select/Read task was the third exposure that subjects had to the 2D selection technique. Also, the same degree of learning effect took place with the 3D interaction

technique between the second and third exposure as occurred with the 2D interaction technique.

One additional fact that further explains the average time difference was that subjects tended to pause and comment on the 2D interaction technique the first time they were required to use it to select an object, generally expressing some confusion. Comments like, “I can’t tell which trailer is the third one from the right,” were common the first time subjects were exposed to the 2D interaction technique. These kinds of comments and some confusion were also expected since a 2D interaction technique does not provide the spatial information that is required to perform a 3D task. However, these comments did not reflect subjects’ lack of understanding about the task requirements. Rather, they were rhetorical in nature, reflecting a recognition of the lack of spatial information on the 2D display.

| | Count | Mean | Std. Dev. | Std. Err. |
|---|---|---|---|---|
| 2D | 27 | 2.074 | 1.752 | .337 |
| 3D | 27 | .074 | .267 | .051 |
| Hybrid | 27 | .074 | .267 | .051 |

| | Mean Diff. | Crit. Diff | P-Value | |
|---|---|---|---|---|
| 2D, 3D | 2.000 | .561 | <.0001 | S |
| 2D, Hybrid | 2.000 | .561 | <.0001 | S |
| 3D, Hybrid | 0.000 | .561 | | |

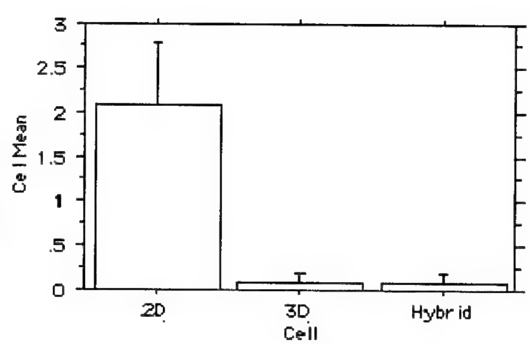



Figure 5.4. Error Results for *Select* Task.

### *b) Accuracy*

The average number of errors committed by each subject in the conduct of the *Select* task even more clearly demonstrates the expected results. A p-value of less than 0.0001 (Figure 5.4) very clearly indicates a statistical significance between the number of errors committed using a 2D interaction technique and the number committed using a 3D interaction technique, despite the 2D interaction technique having a standard deviation of 1.752 and raw data values ranging from 0 to 8 errors. The few errors that occurred when using the 3D interaction technique provided by the 3D and Hybrid interfaces were instances when the subject missed the object with the ray. The misses are attributable mostly to magnetic tracking jitter.

| | Count | Mean | Std. Dev. | Std. Err. |
|---|---|---|---|---|
| 2D | 27 | 1.667 | 1.074 | .207 |
| 3D | 27 | .111 | .320 | .062 |
| Hybrid | 27 | 1.111 | 1.121 | .216 |

| | Mean Diff. | Crit. Diff | P-Value | |
|---|---|---|---|---|
| 2D, 3D | 1.556 | .496 | <.0001 | S |
| 2D, Hybrid | .556 | .496 | .0286 | S |
| 3D, Hybrid | -1.000 | .496 | .0001 | S |

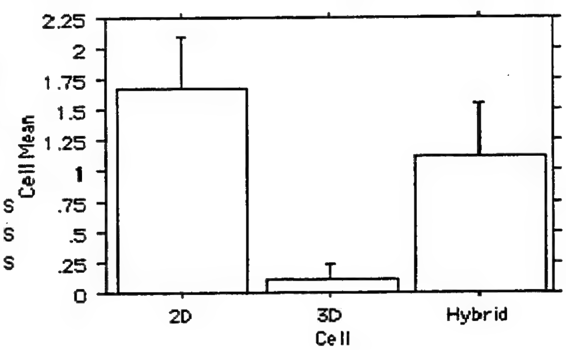



Figure 5.5. Error Results for *Select/Text* Task Combination.

The number of errors committed in performing the Select/Read task also clearly indicates that the performance of a 3D task with a 3D interaction technique is more accurate (Figure 5.5). The results demonstrate a statistically significant difference in the number of errors committed using the Hybrid vs. the 2D interface. This result was

unexpected, since both the 2D and Hybrid interfaces employed the same 2D interaction technique to perform the task. A possible explanation results from considering the amount of exposure to the 2D interaction technique and the object list, the typical strategy employed by test subjects when using the 2D technique, and the ordering of the objects in the scene. Figure 5.6 shows the arrangement of the objects in the scene relative to the list associated with the 2D interaction technique. The numbers in the figure indicate that object's position on the list. Objects 1 – 4 were trucks, and objects 5 – 8 were trailers. As stated earlier, for half of the subjects, the Hybrid interface was their first exposure to the 2D selection technique. For most of those subjects, the first object they were required

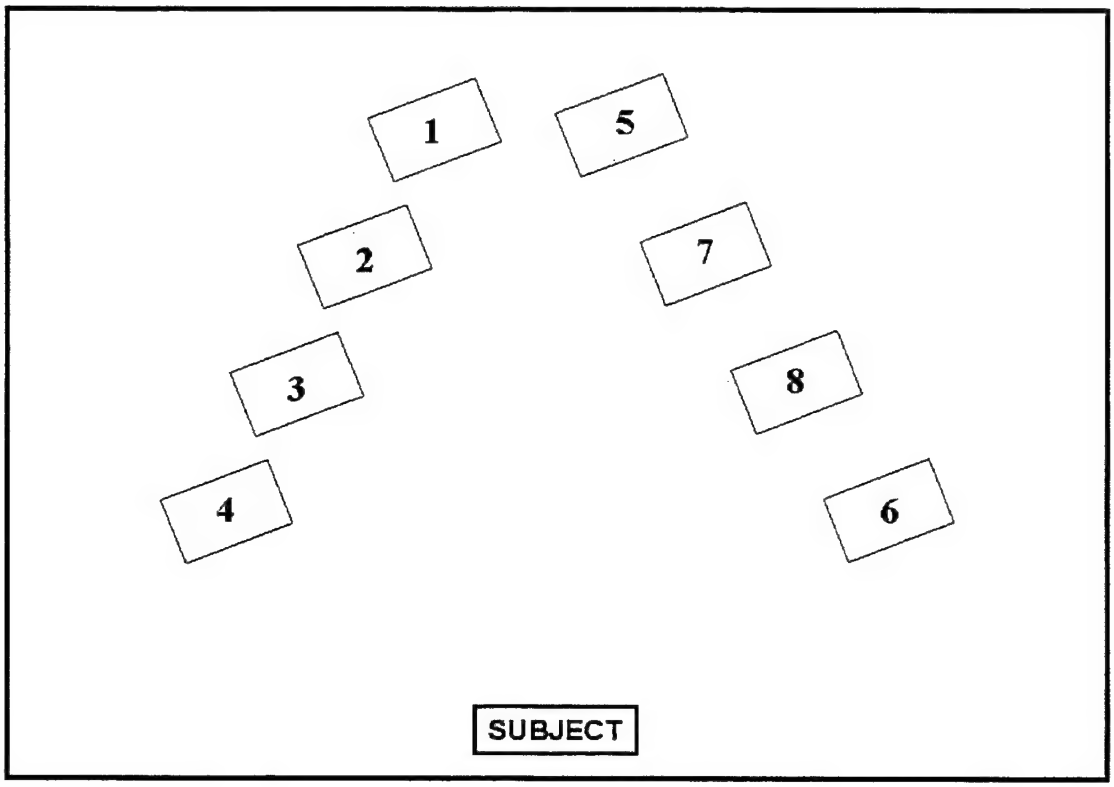



Figure 5.6. Arrangement of Objects in the Scene.

to select was the fourth trailer from the right. Most people in western cultures process items from left to right, and from front to rear. Those subjects who already had experience with the selection list had discovered that the objects were not arranged on the list from left to right. Furthermore, these experienced subjects generally reverted to a trial and error strategy of selecting an object from the list. Since this vehicle was also the fifth object from the left and the fifth item on the list, subjects who were first exposed to the list when selecting object #5 averaged about half as many errors as those who had already been exposed to the list.

### 2. Position Task

The *Position* task provided subjects with a 2D interaction technique while using the 2D interface, and a 3D interaction technique when the 3D and Hybrid interfaces were active. As with the *Select* task, the dimensionality requirements of the task were three-dimensional.

| | Count | Mean | Std. Dev. | Std. Err. |
|---|---|---|---|---|
| 2D | 27 | 112.000 | 54.529 | 10.494 |
| 3D | 27 | 41.556 | 20.999 | 4.041 |
| Hybrid | 27 | 51.185 | 23.724 | 4.566 |

| | Mean Diff. | Crit. Diff | P-Value | |
|---|---|---|---|---|
| 2D, 3D | 70.444 | 19.729 | <.0001 | S |
| 2D, Hybrid | 60.815 | 19.729 | <.0001 | S |
| 3D, Hybrid | -9.630 | 19.729 | .3342 | |

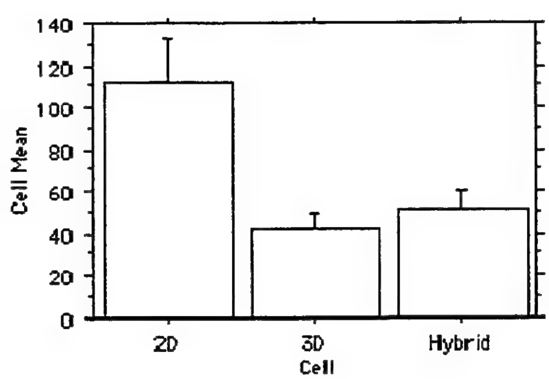



Figure 5.7. Time Results for *Position* Task.

*a) Time*

The performance measurements for the time required to perform the *Position* task clearly show that matching a 3D interaction technique to the 3D requirements of the task resulted in task performance that was faster on average by over a minute. Given a p-value < 0.0001 and a standard deviation that was twice as small as the one associated with the 2D interaction technique (Figure 5.7), there is a statistically significant difference between the time required to accomplish the *Position* task with a 3D technique vs. a 2D technique.

| | Count | Mean | Std. Dev. | Std. Err. |
|---|---|---|---|---|
| 2D | 27 | .852 | 1.027 | .198 |
| 3D | 27 | .259 | .712 | .137 |
| Hybrid | 27 | .481 | .700 | .135 |

| | Mean Diff. | Crit. Diff | P-Value | |
|---|---|---|---|---|
| 2D, 3D | .593 | .448 | .0102 | S |
| 2D, Hybrid | .370 | .448 | .1038 | |
| 3D, Hybrid | -.222 | .448 | .3265 | |

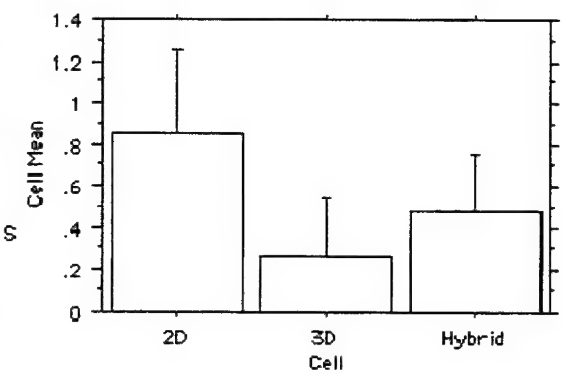



Figure 5.8. Error Results for *Position* Task.

*b) Accuracy*

The average number of errors committed while performing the *Position* task using a 2D technique plainly exceeds the number committed using a 3D technique, as was the case with both the 3D and Hybrid interfaces. The results show statistical significance in the difference between the 2D and 3D interfaces with a p-value = 0.0102 (Figure 5.8). However, despite using the same interaction technique, the Hybrid interface

did not demonstrate the same statistical significance. This can be attributed to one outlying data point. One test subject, while stating that he had no depth perception problems, evidenced extreme difficulty in positioning a trailer behind a truck using the Hybrid interface. The subject in question incorrectly positioned the trailer three times before successfully completing the task. All other subjects committed no more than one error while performing this task. Figure 5.9 provides the updated results when this subject's data was removed from the analysis. These results clearly demonstrate the expected results, producing an average number of errors using the 3D interaction

| | Count | Mean | Std. Dev. | Std. Err. |
|---|---|---|---|---|
| 2D | 27 | .852 | 1.027 | .198 |
| 3D | 27 | .259 | .712 | .137 |
| Hybrid | 26 | .385 | .496 | .097 |

| | Mean Diff. | Crit. Diff | P-Value | |
|---|---|---|---|---|
| 2D, 3D | .593 | .422 | .0066 | S |
| 2D, Hybrid | .467 | .426 | .0321 | S |
| 3D, Hybrid | -.125 | .426 | .5599 | |

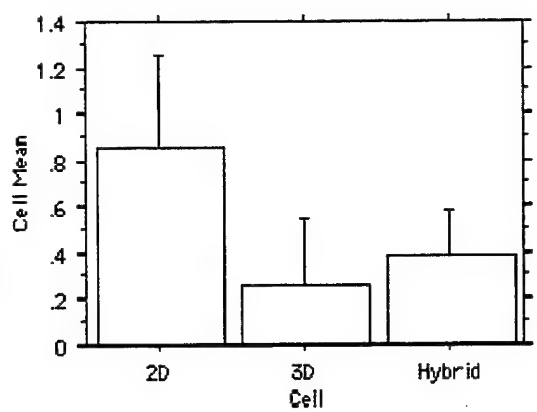



Figure 5.9. Error Results for *Position* Task with Outlying Data Point Removed.

technique that was less than a third of the number of errors committed using the 2D interaction techniques. P-values of 0.0066 and 0.0321 demonstrate the statistical significance of the differences between the 2D interface and the 3D and Hybrid interfaces, respectively.

### 3. Text Tasks

The *Text* tasks were performed by test subjects using a 3D interaction technique with the 3D interface, and a 2D interaction technique with the 2D and Hybrid interfaces. Each subject performed two tasks: the first task required them to read data; the second required them to input numerical data.

#### *a) Time*

By examining the average time required for subjects to read data about a selected object, it is evident that using a 2D technique to display 2D textual data enabled subjects to perform the task more quickly. The 2D and Hybrid interfaces both presented subjects with a 2D technique, whereas the 3D interface employed a 3D technique. There is statistical significance in the difference between the 2D and 3D interaction technique results as evidenced by a p-value < 0.0001 (Figure 5.10).

| | Count | Mean | Std. Dev. | Std. Err. |
|---|---|---|---|---|
| 2D | 27 | 3.000 | 1.301 | .250 |
| 3D | 27 | 11.370 | 5.422 | 1.043 |
| Hybrid | 27 | 3.037 | 1.480 | .285 |

| | Mean Diff. | Crit. Diff | P-Value | |
|---|---|---|---|---|
| 2D, 3D | -8.370 | 1.805 | <.0001 | S |
| 2D, Hybrid | -.037 | 1.805 | .9675 | |
| 3D, Hybrid | 8.333 | 1.805 | <.0001 | S |

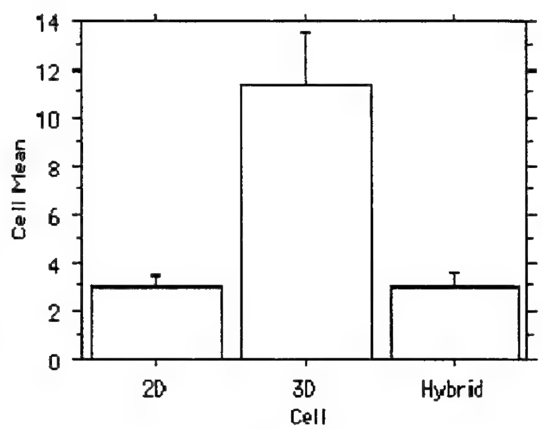



Figure 5.10. Time Results for Read Task.

The results from the textual input task were also very clear (Figure 5.11). The 2D and Hybrid interfaces provided subjects with a 2D interaction technique for

| | Count | Mean | Std. Dev. | Std. Err. |
|---|---|---|---|---|
| 2D | 27 | 15.926 | 5.797 | 1.116 |
| 3D | 27 | 64.000 | 23.344 | 4.492 |
| Hybrid | 27 | 17.185 | 5.731 | 1.103 |

| | Mean Diff. | Crit. Diff | P-Value | |
|---|---|---|---|---|
| 2D, 3D | -48.074 | 7.735 | <.0001 | S |
| 2D, Hybrid | -1.259 | 7.735 | .7467 | |
| 3D, Hybrid | 46.815 | 7.735 | <.0001 | S |

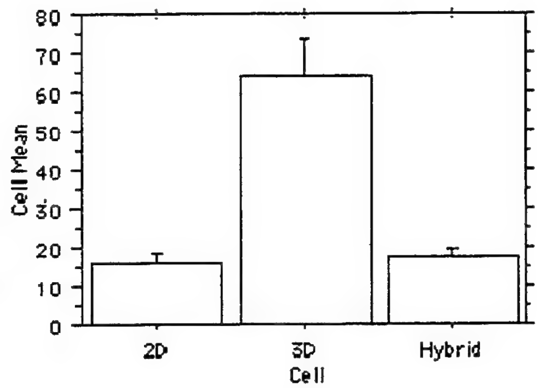



Figure 5.11. Time Results for *Text* Task.

performing an inherently 2D task, whereas the 3D interface required subjects to use a 3D technique for the same task. As a result, subjects took significantly longer to perform the textual input task using the 3D interface. The average time needed to perform the task using the 3D interface was about four times longer than when a 2D interaction technique was provided, and p-values < 0.0001 demonstrate the reliability of the data.

### *b) Accuracy*

As mentioned previously, the number of errors committed in the performance of the Read Data task were insignificant, although it is interesting to note that the one error that occurred in all the iterations of the task occurred when a subject was using a 3D interaction technique to perform the task. Figure 5.12 clearly shows the

statistically significant difference between the number of errors committed using the 2D

| | Count | Mean | Std. Dev. | Std. Err. |
|---|---|---|---|---|
| 2D | 27 | 0.000 | 0.000 | 0.000 |
| 3D | 27 | 1.296 | 1.295 | .249 |
| Hybrid | 27 | .037 | .192 | .037 |

| | Mean Diff. | Crit. Diff | P-Value | |
|---|---|---|---|---|
| 2D, 3D | -1.296 | .410 | <.0001 | S |
| 2D, Hybrid | -.037 | .410 | .8576 | |
| 3D, Hybrid | 1.259 | .410 | <.0001 | S |

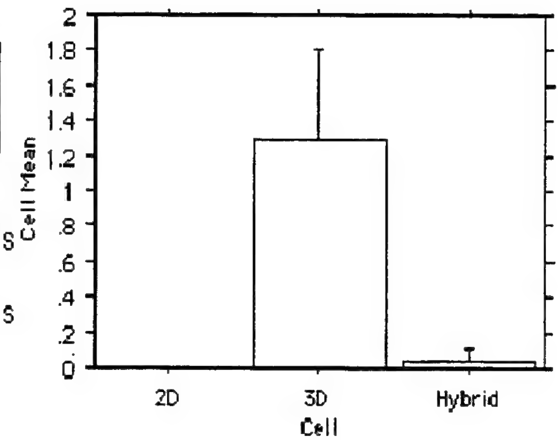



Figure 5.12. Error Results for *Text* Task.

interaction technique provided by the 2D and Hybrid interfaces and those committed using the 3D interaction technique presented by the 3D interface.

## B. PREFERENCE RESULTS

After completing the experiment, all subjects were asked to complete a post-task questionnaire that elicited feedback on subjects' interaction technique preferences for performing each task, as well as a subjective rating of the ease or difficulty with which tasks were performed in each interface. Although user preference does not always correlate with performance, in this case it was expected that the feedback would indicate that subjects preferred to use interaction techniques that matched the dimensionality requirements of the tasks they were required to perform.

### 1. Interface Task Ratings

Experiment subjects were asked to consider each interface separately and rate the technique provided for performing each task. This feedback was compiled by task to provide a comparison of the subjects' ratings by task.

| | Count | Mean | Std. Dev. | Std. Err. |
|---|---|---|---|---|
| 2D | 27 | 2.259 | 1.196 | .230 |
| 3D | 27 | 3.852 | 1.134 | .218 |
| Hybrid 2D | 27 | 2.444 | 1.188 | .229 |
| Hybrid 3D | 27 | 4.148 | .949 | .183 |

| | Mean Diff. | Crit. Diff | P-Value | |
|---|---|---|---|---|
| 2D, 3D | -1.593 | .605 | <.0001 | S |
| 2D, Hybrid 2D | -.185 | .605 | .5452 | |
| 2D, Hybrid 3D | -1.889 | .605 | <.0001 | S |
| 3D, Hybrid 2D | 1.407 | .605 | <.0001 | S |
| 3D, Hybrid 3D | -.296 | .605 | .3337 | |
| Hybrid 2D, Hybrid 3D | -1.704 | .605 | <.0001 | S |

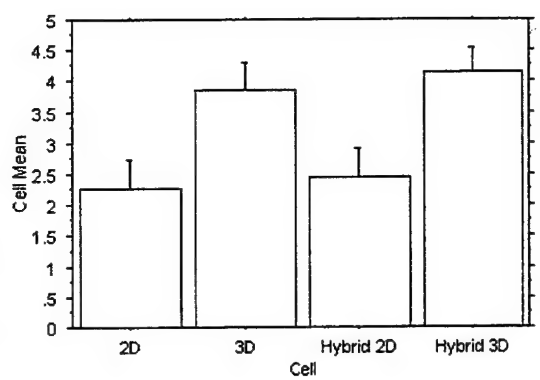



Figure 5.13. Select Task Rating Results.

#### *a) Select Task*

Subjects rated the technique provided for performing a *Select* task with each interface. The Hybrid interface provided both a 2D and a 3D interaction technique, so subjects were asked to rate both techniques when evaluating that interface. The results were clear (Figure 5.13). On average, subjects found the 3D technique to be somewhat easy and the 2D technique to be somewhat difficult. In general, subjects found that using a 3D interaction technique to perform a 3D *Select* task was easier than using a 2D technique.

| | Count | Mean | Std. Dev. | Std. Err. |
|---|---|---|---|---|
| 2D | 27 | 4.852 | .456 | .088 |
| 3D | 27 | 3.630 | 1.115 | .214 |
| Hybrid | 27 | 4.926 | .267 | .051 |

| | Mean Diff. | Crit. Diff | P-Value | |
|---|---|---|---|---|
| 2D, 3D | 1.222 | .386 | <.0001 | S |
| 2D, Hybrid | -.074 | .386 | .7034 | |
| 3D, Hybrid | -1.296 | .386 | <.0001 | S |

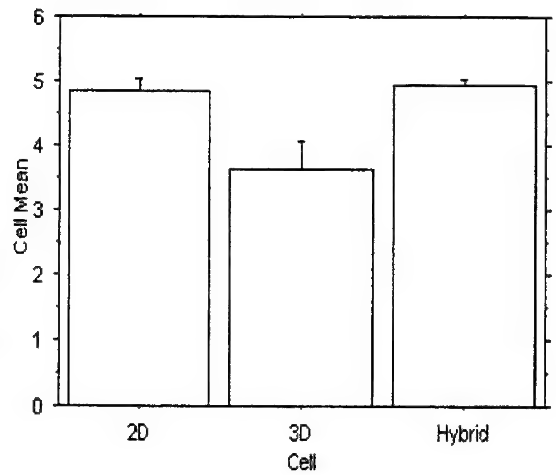



Figure 5.14. Read Task Rating Results.

### *b) Read Task*

Subjects found the Read task to be simple, so average ratings ranged from somewhat easy too easy. As expected, they felt that it was easier to read textual data presented on a 2D interaction device (Figure 5.14). Although many subjects stated that they found the task to be easy regardless of which interface they used, some commented that the 3D presentation made reading more difficult because the letters tended to blend in with the background, making some words more difficult to read. Others commented that they did not like the 3D presentation because it partially obscured their vision of the scene.

| | Count | Mean | Std. Dev. | Std. Err. |
|---|---|---|---|---|
| 2D | 27 | 2.222 | 1.121 | .216 |
| 3D | 27 | 3.074 | 1.466 | .282 |
| Hybrid | 27 | 3.259 | 1.318 | .254 |

| | Mean Diff. | Crit. Diff | P-Value | |
|---|---|---|---|---|
| 2D, 3D | -.852 | .709 | .0192 | S |
| 2D, Hybrid | -1.037 | .709 | .0047 | S |
| 3D, Hybrid | -.185 | .709 | .6047 | |

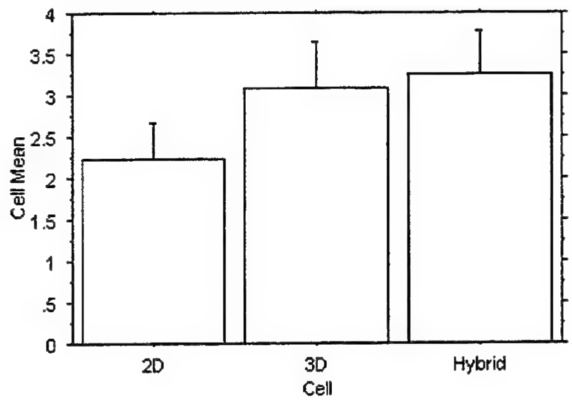



Figure 5.15. Move Task Rating Results.

### c) *Move Task*

Both the 3D and the Hybrid interfaces provided subjects with a 3D interaction technique for performing a *Position* task while the 2D interface presented a 2D interaction technique. Since this was a 3D task, one would expect that subjects would find that a 3D interaction technique was easier to use, and the results demonstrate the veracity of that assumption (Figure 5.15). However, an average rating of slightly more than 3 on this task indicates that subjects did not find the task even somewhat easy. Rather, they were neutral in their rating. Explanatory comments from subjects indicate a possible reason for the neutral rating. Many of them indicated that the jitter they experienced as a result of the magnetic tracking made it difficult to be precise in positioning a trailer behind a truck.

### d) *Assign # Task*

The subjects' rating of this example of an inherently 2D Text task could not be clearer. The Assign # task was rated by all 27 subjects for each interface. Since

both the 2D and Hybrid interfaces provided the same 2D interaction technique to the test subjects, a total of 54 data points were available to assess an overall rating for the 2D interaction technique. Subjects gave the 2D technique the highest rating possible in 50 out of 54 ratings (Figure 5.16). The average rating for the 3D interaction technique fell

| | Count | Mean | Std. Dev. | Std. Err. |
|---|---|---|---|---|
| 2D | 27 | 4.963 | .192 | .037 |
| 3D | 27 | 2.741 | 1.163 | .224 |
| Hybrid | 27 | 4.889 | .320 | .062 |

| | Mean Diff. | Crit. Diff | P-Value | |
|---|---|---|---|---|
| 2D, 3D | 2.222 | .382 | <.0001 | S |
| 2D, Hybrid | .074 | .382 | .7007 | |
| 3D, Hybrid | -2.148 | .382 | <.0001 | S |

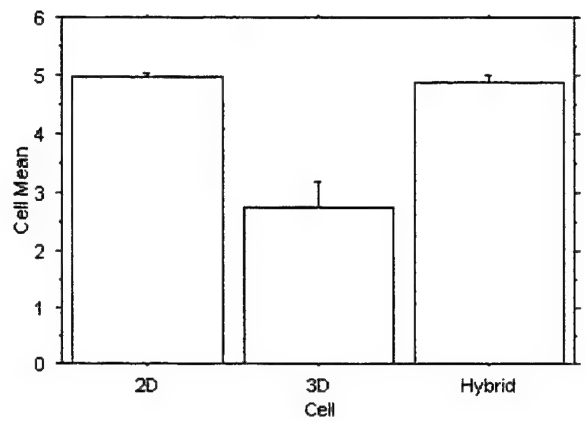


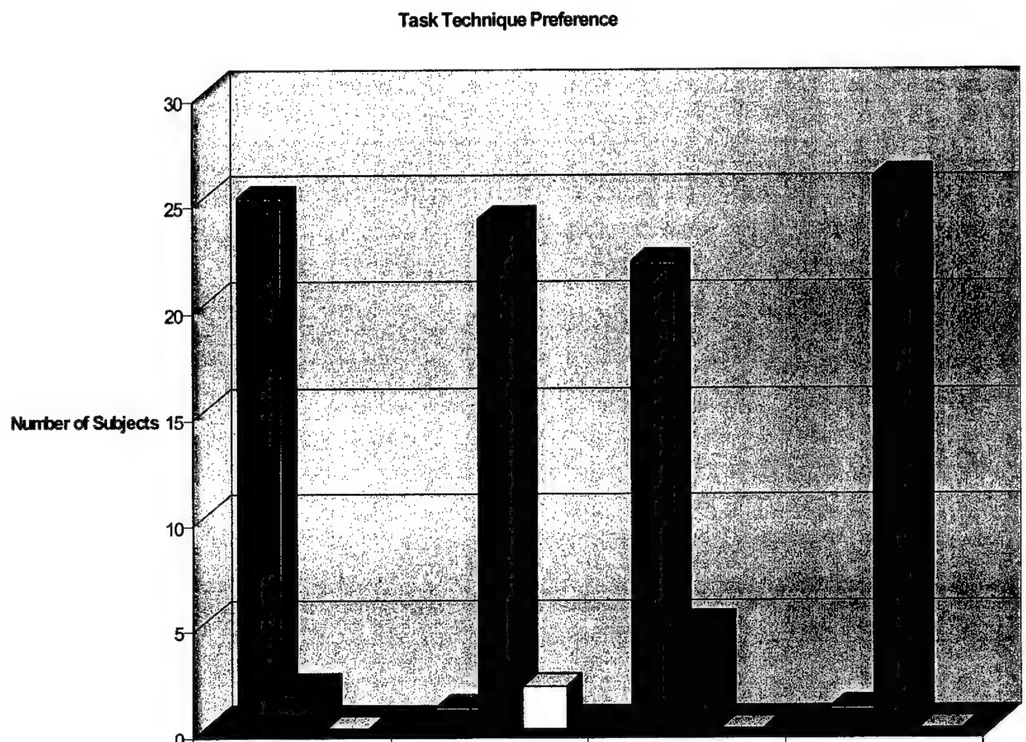

Figure 5.16. Assign # Task Rating Results.

between somewhat difficult and neutral. These results plainly demonstrate the superiority of a 2D technique over a 3D technique for performing this 2D task.

### 2. Overall Interface Rating

The final section of the post-task questionnaire asked the subjects to indicate which technique they preferred to use when performing each of the four tasks. As the data in Figure 5.17 shows, there is a clear trend. When performing the Select and Move tasks, the two tasks whose dimensionality requirements were 3D, subjects preferred to use a 3D interaction technique. When performing the two inherently 2D tasks, Read and Assign #, subjects preferred to use a 2D interaction technique. Subjects were also

asked to indicate which interface they would prefer to use to perform all the tasks. Given




| | Select | Read | Move | Assign # |
|---|---|---|---|---|
| ■ 3D Interaction Technique | 25 | 1 | 22 | 1 |
| ■ 2D Interaction Technique | 2 | 24 | 5 | 26 |
| □ No Preference | 0 | 2 | 0 | 0 |

Task

Figure 5.17. Task Technique Preference.

all the previous preference indicators and the performance results, it was no surprise to find that the majority of the subjects preferred the Hybrid interface, as it was the only interface that provided interaction techniques whose dimensionality properties matched the requirements of the associated tasks. The Hybrid interface was preferred by five times as many subjects as the 2D interface and by nearly 7 times as many subjects as the 3D interface (Figure 5.18).

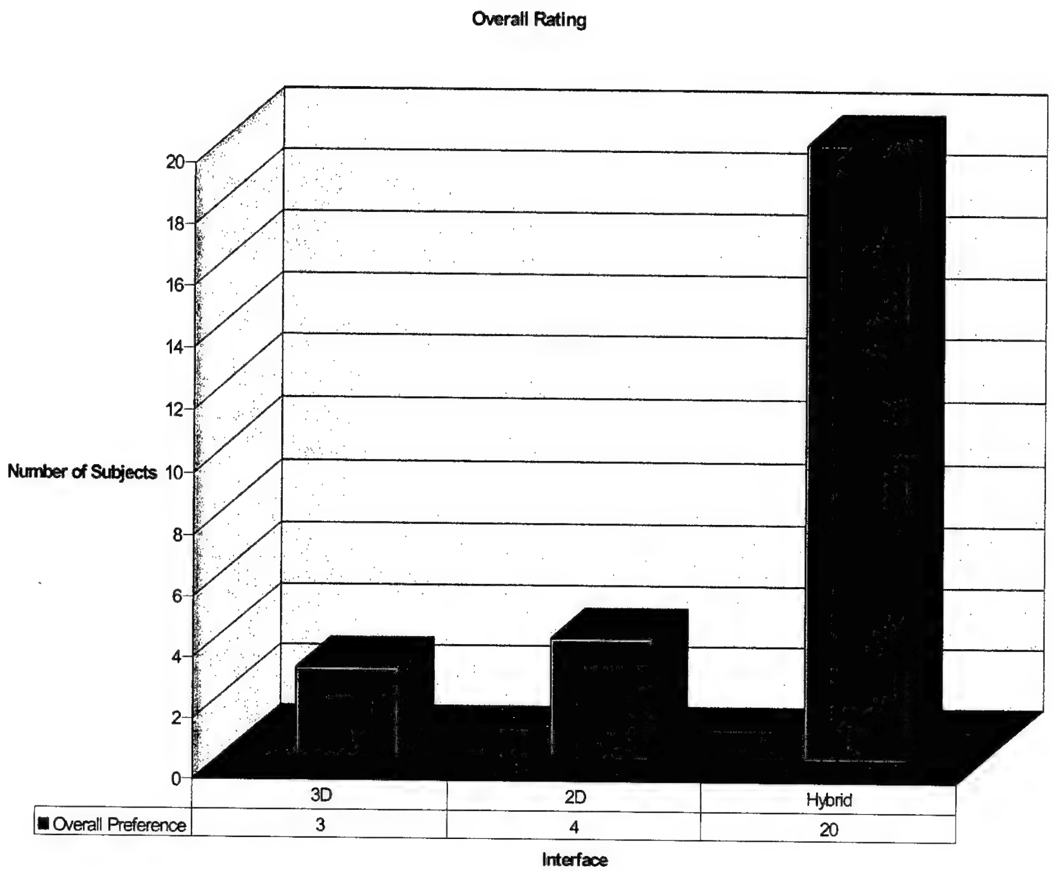



Figure 5.18. Overall Interface Preference.

## C. DISCUSSION

This experiment was designed to show that matching the dimensionality of task requirements to interaction techniques will improve performance on tasks as opposed to when a mismatch occurs. The results of both performance and preference measurements clearly prove this hypothesis. The 3D tasks, Select and Move, were performed faster and with a greater degree of accuracy when subjects used a 3D interaction technique. Subjects felt the 3D interaction technique was easier than the 2D technique for

performing these tasks, and indicated that they preferred matching 3D techniques to tasks with 3D requirements. Comments associated with subjects' ratings and preferences indicated that although the 3D interaction technique was preferred for 3D tasks, the jitter that resulted from the use of magnetic tracking prevented the 3D techniques from being rated as easy. This technological artifact drove average ratings of "somewhat easy" for the Select task and neutral for the Move task.

The 2D interaction techniques associated with the inherently 2D Assign # and Read tasks were very clearly favored by the majority of subjects. Both techniques were considered easy to use and were preferred by most subjects over the 3D interaction techniques associated with the Assign # and Read tasks. Their performance agreed with their preferences — subjects accomplished both tasks more quickly and with fewer errors when using 2D interaction techniques.

THIS PAGE INTENTIONALLY LEFT BLANK

# VI. CONCLUSIONS AND FUTURE WORK

This chapter provides a synopsis of the material covered throughout this thesis and explores other areas, related to this work, where further research could prove to be beneficial.

## A. EFFECTS OF DIMENSIONALITY MATCHING

3D tasks in virtual environment applications are accomplished well using 3D interaction techniques that enable users to select, position, and orient objects with ease. However, not all tasks that need to be performed in a virtual environment are 3D. Unfortunately, most VE applications are not designed to provide users with a method for performing 2D tasks. Alternatively, they provide users with 3D techniques for performing all tasks in the environment.

It seems intuitive that matching the dimensionality of task requirements to interaction techniques will improve performance on tasks as opposed to when a mismatch occurs. Prior to this work, no formal testing had ever been performed, and thus no empirical data existed to demonstrate the validity of this theory. This thesis has attempted to fill both those voids by conducting an experiment and providing data that demonstrates the improved performance that results from matching the dimensionality of task requirements to interaction techniques.

Furthermore, this thesis demonstrates that it is possible to perform both 2D and 3D tasks in a virtual environment without sacrificing the functionality of one for the other. Numerous techniques exist for performing 2D or 3D tasks. However, these

techniques need not be mutually exclusive. One can, and should be able to perform both 2D and 3D tasks in a virtual environment using a device or devices that are designed to accommodate the dimensionality requirements of the task being performed.

### 1. Faster Performance

The first experimental performance measure recorded the amount of time subjects required to perform a variety of tasks using 2D and 3D interaction techniques. In keeping with the hypothesis of this thesis, it was expected that test subjects would perform most quickly when the dimensionality of the technique they were required to use matched the dimensionality requirements of the tasks they were being asked to perform. The results showed that was indeed the case, as subjects performed the 3D Select and Move tasks best when provided with a 3D interaction technique, and were fastest in accomplishing the 2D Read and Assign # tasks when using a 2D technique.

### 2. More Accurate Performance

Accuracy was the second performance measurement, comparing the number of errors that occurred when the dimensionality of the task and technique were mismatched against the instances when they were not. Again, as expected, test subjects were most accurate in accomplishing tasks when the dimensionality of the technique they were required to use matched the dimensionality of the task being performed.

### 3. Preferred Configuration

A final set of data examined after conducting this experiment looked at subjects' preferences. Although user preference is not always an accurate indicator of performance, the results of this experiment showed that both performance and preference

demonstrated the same trends. Subjects found that tasks were most easily performed when the dimensionality of the interaction technique matched the dimensionality of the task. Furthermore, their overall preference indicated that they favored an interface that provided them with the means to perform both 2D and 3D tasks without forcing them to mismatch the dimensionality of tasks and techniques.

## B. FUTURE WORK

Proving that matching the dimensionality of task requirements to interaction techniques improves performance is an important step for VE application design. However, the resolution of this hypothesis does not indicate that no further research and experimentation are needed in this area.

### 1. Overall Performance Comparison

The experiment performed in this thesis can be extended by comparing the overall performance of groups of test subjects given different conditions. In a new experiment, subjects can be divided into three groups, with all groups performing a series of 2D and 3D tasks. The first group would use only 3D interaction techniques, the second group would use 2D interaction techniques, and the third group would use properly task-matched interaction techniques. Performance measurements would again examine the speed and accuracy with which subjects accomplish the tasks, this time taking cumulative measurements of time and errors. The results, if similar to those of the experiment in this thesis, would provide further proof of the validity of these results.

### 2. Separability and Integrality

Another issue not considered by this thesis, but relevant given the clear line drawn between 2D and 3D tasks, is the issue of separability vs. integrality. Certain types of tasks are best performed when separated or performed using distinct techniques or devices, while others are accomplished more quickly and accurately when integrated. Determining how tasks fall into these two categories is an important issue to consider given the introduction of the makeshift hybrid device used in conducting the experiment associated with this thesis. Such a hybrid device enables 2D and 3D tasks to be integrated. If, however, the tasks are best performed when separated, it is important to provide users with devices and techniques that enable them to accomplish the tasks separately while still matching the dimensionality requirements of tasks to the techniques provided.

### 3. Other 2D Devices

This experiment was conducted using a heavy, hand-held tablet that would be difficult to use for an extended time in a VE. Future research should look at other options such as the use of a wearable computer or a Personal Digital Assistant (PDA) such as a PalmPilot. These devices are already capable of performing many robust applications and are widely used in business, making them an excellent candidate for incorporation into future VE applications as 2D and/or hybrid interaction devices.

## APPENDIX A – BENCHMARK TASK LIST

**Benchmark #1 (measure task performance time, number of errors)**

- Perform Select, Read, Move, and Assign# tasks using 3D interaction techniques.

**Benchmark #2 (measure task performance time, number of errors)**

- Perform Select, Read, Move, and Assign# tasks using 2D interaction techniques.

**Benchmark #3 (measure task performance time, number of errors)**

- Perform Select, Read, Move, and Assign# tasks using hybrid interaction techniques.

THIS PAGE INTENTIONALLY LEFT BLANK

## APPENDIX B – EXPERIMENT OVERVIEW

Virtual Environments are used in research, training, and manufacturing. The wide range of tasks that need to be performed in a Virtual Environment require a wide range of interaction techniques and devices. This usability test is designed to test a variety of interaction techniques in order to determine what types of interaction techniques are best suited for tasks that may commonly be performed in a Virtual Environment. Your task is to perform a series of four tasks using a variety of techniques as directed.

You will be asked to perform the following tasks:

- Select a truck/trailer in the Virtual Environment.
- Read data about the selected vehicle.
- Move a trailer from its starting location to a position behind a truck.
- Assign a number to one of the trucks.

THIS PAGE INTENTIONALLY LEFT BLANK

## APPENDIX C – PARTICIPANT CONSENT FORM

1. **Introduction.** You are invited to participate in a study of interaction techniques in virtual environments. With information gathered from you and other participants, we hope to discover insight on interaction techniques used to perform tasks in virtual environments that require both two and three-dimensional interaction. We ask you to read and sign this form indicating that you agree to be in the study. Please ask any questions you may have before signing.

2. **Background Information.** The Naval Postgraduate School NPSNET Research Group is conducting this study.

3. **Procedures.** If you agree to participate in this study, the researcher will explain the tasks in detail. There will be one session lasting approximately 30 minutes, during which you will be expected to accomplish a number of interactive tasks related to the selection and manipulation of objects as well as textual display and entry.

4. **Risks and Benefits.** This research involves no risks or discomforts greater then those encountered in using a hand-held computer. The benefits to the participants are exposure to a variety of interaction techniques that can be used in a virtual environment and contributing to current research in human-computer interaction.

5. **Compensation.** No tangible reward will be given. A copy of the results will be available to you upon request at the conclusion of the experiment.

6. **Confidentiality.** The records of this study will be kept confidential. No information will be publicly accessible which could identify you as a participant.

7. **Voluntary Nature of the Study.** If you agree to participate, you are free to withdraw from the study at any time without prejudice. You will be provided a copy of this form for your records.

8. **Points of Contact.** If you have any further questions or comments after the completion of the study, you may contact the research supervisor, Dr. Rudolph P. Darken (831) 656-4072 darken@nps.navy.mil.

9. **Statement of Consent.** I have read the above information. I have asked all question and have had my questions answered. I agree to participate in this study.

---------------------------------------------- --------------------------

Participant's Signature Date

---------------------------------------------- --------------------------

Researcher's Signature Date

THIS PAGE INTENTIONALLY LEFT BLANK

# APPENDIX D - MINIMAL RISK CONSENT STATEMENT

## NAVAL POSTGRADUATE SCHOOL, MONTEREY, CA 93943
## MINIMAL RISK CONSENT STATEMENT

Participant: VOLUNTARY CONSENT TO BE A RESEARCH PARTICIPANT IN: Virtual Environments and Navigation in Natural Environments

1. I have read, understand and been provided "Information for Participants" that provides the details of the below acknowledgments.

2. I understand that this project involves research. An explanation of the purposes of the research, a description of procedures to be used, identification of experimental procedures, and the extended duration of my participation have been provided to me.

3. I understand that this project does not involve more than minimal risk. I have been informed of any reasonably foreseeable risks or discomforts to me.

4. I have been informed of any benefits to me or to others that may reasonably be expected from the research.

5. I have signed a statement describing the extent to which confidentiality of records identifying me will be maintained.

6. I have been informed of any compensation and/or medical treatments available if injury occurs and is so, what they consist of, or where further information may be obtained.

7. I understand that my participation in this project is voluntary; refusal to participate will involve no penalty or loss of benefits to which I am otherwise entitled. I also understand that I may discontinue participation at any time without penalty or loss of benefits to which I am otherwise entitled.

8. I understand that the individual to contact should I need answers to pertinent questions about the research is Rudy Darken, Ph.D., Principal Investigator, and about my rights as a research participant or concerning a research related injury is the Modeling Virtual Environments and Simulations Chairman. A full and responsive discussion of the elements of this project and my consent has taken place.

Medical Monitor: Flight Surgeon, Naval Postgraduate School

________________________________________

Signature of Principal Investigator Date

________________________________________

Signature of Volunteer Date

________________________________________

Signature of Witness Date

THIS PAGE INTENTIONALLY LEFT BLANK

## APPENDIX E – PRIVACY ACT STATEMENT

## NAVAL POSTGRADUATE SCHOOL, MONTEREY, CA 93943
## PRIVACY ACT STATEMENT

1. Authority: Naval Instruction

2. Purpose: Spatial cognition data will be collected to enhance knowledge, and to develop tests, procedures, and equipment to improve the development of Virtual Environments.

3. Use: Spatial cognition data will be used for statistical analysis by the Departments of the Navy and Defense, and other U.S. Government agencies, provided this use is compatible with the purpose for which the information was collected. Use of the information may be granted to legitimate non-government agencies or individuals by the Naval Postgraduate School in accordance with the provisions of the Freedom of Information Act.

4. Disclosure/Confidentiality:

   a. I have been assured that my privacy will be safeguarded. I will be assigned a control or code number, which thereafter will be the only identifying entry on any of the research records. The Principal Investigator will maintain the cross-reference between name and control number. It will be decoded only when beneficial to me or if some circumstances, which are not apparent at this time, would make it clear that decoding would enhance the value of the research data. In all cases, the provisions of the Privacy Act Statement will be honored.

   b. I understand that a record of the information contained in this Consent Statement or derived from the experiment described herein will be retained permanently at the Naval Postgraduate School or by higher authority. I voluntarily agree to its disclosure to agencies or individuals indicated in paragraph 3 and I have been informed that failure to agree to such disclosure may negate the purpose for which the experiment was conducted.

   c. I also understand that disclosure of the requested information, including my Social Security Number, is voluntary.

Signature of Volunteer   Name, Grade/Rank (if applicable)   DOB   SSN   Date

Signature of Witness   Date

THIS PAGE INTENTIONALLY LEFT BLANK

## APPENDIX F – DEMOGRAPHIC QUESTIONNAIRE

Name:________________________________ Rank:______________

Years of Service: __________ Curriculum:__________________________

Years of using computer: ________________

Do you own a computer: ________________

Do you have any known depth perception problems? ________________

If so, please explain ______________________________________________

______________________________________________________________

______________________________________________________________

______________________________________________________________

______________________________________________________________

Do you have any past experience using a virtual environment?______________

If so, please explain ______________________________________________

______________________________________________________________

______________________________________________________________

______________________________________________________________

______________________________________________________________

<u>Rate yourself from 1 (Novice) to 3 (Expert) on the following:</u>

Computer expertise: ______

Understanding of Windows interfaces: ______

Ability to work in a multi-window environment: ______

THIS PAGE INTENTIONALLY LEFT BLANK

## APPENDIX G – INTERACTION INTERFACE HELP PAGE

This experiment requires the user to perform an identical series of tasks using three different implementations; one consisting of primarily three-dimensional interaction techniques, one consisting of primarily two-dimensional interaction techniques, and a third consisting of a combination of two and three-dimensional techniques. Helpful hints about the interface are as follows:

### 3D Tasks:

- When performing the "Select", "Move", or "Assign#" tasks, it is important to make sure that you place the cursor in the center of the tablet display. Leaving the cursor on one of the buttons will only call that function again and will not register a selection you are trying to make in the scene.

- Due to network latency, you must hold the ray on the object you are selecting until either the ray disappears (Select and Move tasks) or the number appears on the green truck (Assign# task).

- When performing the "Move" task, hold the Mouse Pen button down and move the trailer so that the front face of the trailer intersects the back face of the truck cab.

### 2D Tasks:

- When performing the "Select" task, you must select the truck or trailer from the pull down list and then tap the "Select" button.

- When performing the "Move" task, use the slide bars to move the trailer so that the front face of the trailer intersects the back face of the truck cab.

- When assigning a number, a dialog box will appear with a text field and a "Done" button. To enter a number, click on the "A" in the text field and a screen keyboard will appear. Enter the number using the keyboard and click on "Done." The number you entered will appear in the text field followed by an "A." Click "Done" in the dialog box. The number will appear on the truck as you typed it (without the "A").

## Hybrid Tasks:

- The "Select" task can be performed in one of two ways. An object can be selected by simply clicking on the "Select" button and picking an object using the same technique used in the 3D Tasks. NOTE: In order for this to work, the text field next to the button must say "Select a Truck or Trailer." An item can also be selected by picking it from the pull-down list and then clicking on the "Select" button.

- When performing the "Select" (using the 3D selection technique) or "Move" tasks, it is important to make sure that you place the cursor in the center of the tablet display. Leaving the cursor on one of the buttons will only call that function again and will not register a selection you are trying to make in the scene.

- The "Assign#" task uses a technique identical to the one used in the 2D Tasks.

## APPENDIX H – EXPERIMENT TASKS

This experiment requires the user to perform an identical series of tasks using three different implementations; one consisting of primarily three-dimensional interaction techniques, one consisting of primarily two-dimensional interaction techniques, and a third consisting of a combination of two and three-dimensional techniques. Instructions for using the interface are as follows:

### <u>3D Tasks:</u>

- Tap the stylus on the "3D Tasks" tab to bring it to the front of the display.
- Tap the stylus on the "Select Object" button and then use the Mouse Pen end of the stylus to select the _______ truck from the left in the scene. A message will appear on the tablet indicating which truck has been selected.
- Tap the stylus on the "Read Data" button on the tablet display. Read the data that is displayed about the _____ truck. Tell the observer the GVWR of the truck.
- Perform the same steps with the _____ trailer from the right, first selecting the _____ trailer and then reading the data about it. Tell the observer the axle rating of the rear axle.
- Tap the stylus on the "Move Object" button and then use the Mouse Pen end of the stylus to move the _______ trailer to the _______ truck. When you release the Mouse Pen button, the trailer will automatically hitch to the truck, provided the front face of the trailer is intersecting the back face of the truck cab.
- Tap the stylus on the "Assign #" button. A series of number buttons will appear in the environment. Use the Mouse Pen to enter the year you were born. This number will be displayed on the green truck as you enter each number. Once you have successfully entered the number, select the "Done" button and the number buttons will disappear.

## 2D TASKS:

- Tap the stylus on the "2D Tasks" tab to bring it to the front of the display.

- Select the _____ truck from the left using the pull-down list beside the "Select Object" button and then tap the stylus on the "Select Object" button. A message will appear confirming your selection.

- Tap the stylus on the "Read Data" button and read the data about the _____ truck displayed on the tablet. Tell the observer the GVWR of the truck.

- Perform the same steps with the _____ trailer from the right, first selecting the _____ trailer and then reading the data about it. Tell the observer the axle rating of the rear axle.

- Tap the stylus on the "Move Object" button. Use the slide bars to position the _______ trailer behind the _______ truck. When you have the trailer positioned behind the truck, press the "Done" button. The dialog box will disappear and the trailer will automatically connect to the truck, provided it is positioned correctly.

- Tap the stylus on the "Assign#" button. Enter the year you were born and click "Done." The dialog box will disappear and the number will appear displayed on the side of the green truck.

## HYBRID TASKS:

- Tap the stylus on the "Hybrid Tasks" tab to bring it to the front of the display.

- Tap the stylus on the "Select Object" button and then pick the _______ truck from the left in the scene using the Mouse Pen. The name of the truck will appear in the selection field next to the "Select Object" button.

- Tap the stylus on the "Read Data" button and read the data on the display about the truck. Tell the observer the GVWR of the truck.

- Use the pull down list and "Select Object" button to select the _______ trailer from the right. A message will appear acknowledging which _______ trailer was selected. Read the data about the trailer. Tell the observer the axle rating.

- Tap the stylus on the "Move Object" button and then use the Mouse Pen end of the stylus to move the _______ trailer to the _______ truck. When you release the Mouse Pen button, the trailer will automatically hitch to the truck, provided the front face of the trailer is intersecting the back face of the truck cab.

- Tap the stylus on the "Assign#" button. Enter the year you were born and click "Done." The dialog box will disappear and the number will appear displayed on the side of the green truck.

THIS PAGE INTENTIONALLY LEFT BLANK

## APPENDIX I – POST TASK QUESTIONNAIRE

**3D Tasks:**

1. Rate the technique used to select a vehicle using the 3D interface.

| Difficult | Somewhat Difficult | Neutral | Somewhat Easy | Easy |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |

Why did you assign this rating?

____________________________________________________________

____________________________________________________________

____________________________________________________________

2. Rate the technique used to read data about the selected vehicle using the 3D interface.

| Difficult | Somewhat Difficult | Neutral | Somewhat Easy | Easy |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |

Why did you assign this rating?

____________________________________________________________

____________________________________________________________

____________________________________________________________

3. Rate the technique used to move objects in the scene using the 3D interface.

| Difficult | Somewhat Difficult | Neutral | Somewhat Easy | Easy |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |

Why did you assign this rating?

______________________________________________________________

______________________________________________________________

______________________________________________________________

4. Rate the technique used to assign a number to the green vehicle using the 3D interface.

| Difficult | Somewhat Difficult | Neutral | Somewhat Easy | Easy |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |

Why did you assign this rating?

______________________________________________________________

______________________________________________________________

______________________________________________________________

**2D Tasks:**

1. Rate the technique used to select a vehicle using the 2D interface.

| Difficult | Somewhat Difficult | Neutral | Somewhat Easy | Easy |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |

Why did you assign this rating?

________________________________________

________________________________________

________________________________________

2. Rate the technique used to read data about the selected vehicle using the 2D interface.

| Difficult | Somewhat Difficult | Neutral | Somewhat Easy | Easy |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |

Why did you assign this rating?

________________________________________

________________________________________

________________________________________

3. Rate the technique used to move objects in the scene using the 2D interface.

| Difficult | Somewhat Difficult | Neutral | Somewhat Easy | Easy |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |

Why did you assign this rating?

______________________________________________

______________________________________________

______________________________________________

4. Rate the technique used to assign a number to the green vehicle using the 2D interface.

| Difficult | Somewhat Difficult | Neutral | Somewhat Easy | Easy |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |

Why did you assign this rating?

______________________________________________

______________________________________________

______________________________________________

**Hybrid Tasks:**

1. Rate the technique used to select a vehicle using the Hybrid interface.

| Difficult | Somewhat Difficult | Neutral | Somewhat Easy | Easy |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |

Why did you assign this rating?

______________________________________________________________

______________________________________________________________

______________________________________________________________

2. Rate the technique used to read data about the selected vehicle using the Hybrid interface.

| Difficult | Somewhat Difficult | Neutral | Somewhat Easy | Easy |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |

Why did you assign this rating?

______________________________________________________________

______________________________________________________________

______________________________________________________________

3. Rate the technique used to move objects in the scene using the Hybrid interface.

| Difficult | Somewhat Difficult | Neutral | Somewhat Easy | Easy |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |

Why did you assign this rating?

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

4. Rate the technique used to assign a number to the green vehicle using the Hybrid interface.

| Difficult | Somewhat Difficult | Neutral | Somewhat Easy | Easy |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |

Why did you assign this rating?

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

**General Questions:**

Of the three interfaces (2D, 3D, and Hybrid), which one did you prefer to use to perform all the interaction tasks?

______________________________________________________________________

Why?

______________________________________________________________________

______________________________________________________________________

______________________________________________________________________

______________________________________________________________________

2. Compare the 2D, 3D, and Hybrid interaction techniques:

A. Which technique or techniques were the easiest to use for selecting object in the scene?

______________________________________________________________________

What made that technique easier than the others?

______________________________________________________________________

______________________________________________________________________

______________________________________________________________________

B. Which technique or techniques did you prefer to use for reading information about the selected objects?

______________________________________________________________________

What made that technique easier than the others?

______________________________________________________________________

______________________________________________________________________

______________________________________________________________________

C. Which technique or techniques were the easiest to use for moving objects in the scene?

______________________________________________________________

What made that technique easier than the others?

______________________________________________________________

______________________________________________________________

______________________________________________________________

D. Which technique or techniques were the easiest to use for assigning a number to the green truck?

______________________________________________________________

What made that technique easier than the others?

______________________________________________________________

______________________________________________________________

______________________________________________________________

3. If you were to pick the techniques that you liked that best and combine then into one interface, what techniques would you use?

______________________________________________________________

______________________________________________________________

What would the interface look like?

______________________________________________________________

______________________________________________________________

______________________________________________________________

## LIST OF REFERENCES


Bochenek, G., "CAVE Automatic Virtual Environment." [http://www.tacom.army.mil/tardec/nac/projects/cave.pdf]. May 2000.

Bowman, D. A., and Hodges, L. F., "An Evaluation of Techniques for Grabbing and Manipulating Remote Objects in Immersive Virtual Environments," *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, pp. 35-38, April 1997.

Christianson, B. K., and Kimsey, A. J., *Comparison of Vega™ and Java3D™ in a Virtual Environment Enclosure*, Master's Thesis, Naval Postgraduate School, Monterey, California, March 2000.

Cutler, L. D., B. Fr★hlich, and P. Hanrahan, "Two-Handed Direct Manipulation on the Responsive Workbench," *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, pp. 107-114, April 1997.

Foley, J. D., V. L. Wallace, and P. Chan, "The Human Factors of Computer Graphics Interaction Techniques," IEEE Computer Graphics and Application, 4(11), Nov. 1984, pp. 13-48.

Lindeman, R. W., J. L. Sibert, and J. K. Hahn, "Hand-Held Windows: Towards Effective 2D Interaction in Immersive Virtual Environment," *Virtual Reality, 1999. Proceedings.*, pp. 205-212, May 1999.

Lindeman, R. W., J. L. Sibert, and J. K. Hahn, "Towards Usable VR: An Empirical Study of User Interfaces for Immersive Virtual Environments," *Proceedings of the CHI '99 Conference on Human Factors in Computing Systems*, pp. 64-71, May 1999.

Pierce, J. S., A. Forsberg, M. J. Conway, S. Hong, R. Zeleznik, and M. R. Mine, "Image Plane Interaction Techniques In 3D Immersive Environments," *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, pp. 39-43, April 1997.

Poupyrev, I., M. Billinghurst, S. Weghorst, and T. Ichikawa, "The Go-Go Interaction Technique: Non-linear Mapping for Direct Manipulation in VR," *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*, pp. 79-80, November 1996.

Poupyrev, I. N. Tomokazu, and S. Weghorst, "Virtual Notepad: Handwriting in Immersive VR," *Proceedings of Virtual Reality Annual International Symposium 1998*, pp. 126-132, March 1998.

Schlager, M. S., "An Approach For Designing Virtual Environment Training Systems," *Proceedings of the CHI '94 Conference Companion on Human Factors in Computing Systems*, pp. 317-318, April 1994.

Schmalstieg, D., L. M. Encarna◇◈o, and Z. Szalav·ri, "Using Transparent Props For Interaction With The Virtual Table," *Proceedings of the 1999 Symposium on Interactive 3D Graphics*, pp. 147-153, 232, April 1997.

Sowizral, H., I. G. Angus, S. Bryson, S. Haas, M. R. Mine, and R. Pausch, "Performing Work Within Virtual Environments," *Proceedings of the 22nd Annual ACM Conference on Computer Graphics*, p. 497, August 1995.

*The American Heritage Dictionary of the English Language*, Third Edition, Houghton Mifflin Company, 1996.

UNC Chapel Hill Computer Science Technical Report TR95-018, *Virtual Environment Interaction Techniques*, by M. R. Mine, pp. 1-18, 5 May 1995.

UNC Chapel Hill Computer Science Technical Report TR96-029, *Working in a Virtual World: Interaction Techniques Used in the Chapel Hill Immersive Modeling Program*, by M. R. Mine, pp. 1-14, 1 August 1996.

Watsen, K., R. P. Darken, and M. V. Capps, "A Handheld Computer as an Interaction Device to a Virtual Environment," paper presented at the 3rd International Immersive Projection Technology Workshop (IPTW'99), Stuttgart, Germany, 1999.

*Webster's Revised Unabridged Dictionary*, MICRA, Inc., 1998.

Wloka, M. M., and Greenfield, E., "The Virtual Tricorder: A Uniform Interface for Virtual Reality," *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*, pp. 39-40, November 1995.

Zeleznik, R. C., A. S. Forsberg, and P. S. Strauss, "Two Pointer Input For 3D Interaction," *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, pp. 115-120, 232, April 1997.

## INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center .......................................................................... (2)
   8725 John J. Kingman Road, Ste 0944
   Ft. Belvoir, VA 22060-6218

2. Dudley Knox Library.................................................................................................. (2)
   Naval Postgraduate School
   411 Dyer Road
   Monterey, CA 93943-5101

3. Chair, Computer Science Department.................................................................... (1)
   Naval Postgraduate School
   Code CS
   Monterey, CA 93943

4. Professor Michael Zyda.............................................................................................. (1)
   Naval Postgraduate School
   Code CS/Zk
   Monterey, CA 93943

5. Assistant Professor Rudolph P. Darken ...................................................................... (3)
   Naval Postgraduate School
   Code CS/Dr
   Monterey, CA 93943

6. Research Assistant Professor Michael V. Capps...................................................... (3)
   Naval Postgraduate School
   Code CS/Cm
   Monterey, CA 93943

7. Dr. Mark Mine............................................................................................................ (1)
   Walt Disney Imagineering VR Studio
   1201 Grand Central
   Glendale, CA 91201

8. Richard S. Durost ....................................................................................................... (1)
   804 S Lake St.
   Lake Mills, IA 50450